

JAERI-M

5590

原子炉核計算コードにおける反復解法の
収束加速アルゴリズム

1974年3月

中原 康明・藤村統一郎

この報告書は、日本原子力研究所が JAERI-M レポートとして、不定期に刊行している研究報告書です。入手、複製などのお問い合わせは、日本原子力研究所技術情報部（茨城県那珂郡東海村）あて、お申しこしください。

JAERI-M reports, issued irregularly, describe the results of research works carried out in JAERI. Inquiries about the availability of reports and their reproduction should be addressed to Division of Technical Information, Japan Atomic Energy Research Institute, Tokai-mura, Naka-gun, Ibaraki-ken, Japan.

原子炉核計算コードにおける反復解法の収束加速アルゴリズム

日本原子力研究所東海研究所原子炉工学部

中原 康明・藤村 統一郎

(1974年2月4日受理)

一般に比較的広く用いられている中性子輸送SNコード及び中性子拡散コードで用いられている有限差分方程式の反復解法の収束加速法についての統一的なサーベイを行った。また既存のコードで未だ用いられていない新しい加速法に関する最近の研究についてもサーベイした。さらに公表されている数値実験例をもとにして各種加速法についての評価を行った。

これらの検討結果から粗メッシュ再釣合法, COP法, 線優緩和法, チェビシェフ加速法及び改良型ウィーラント法が有望な加速法であるとの結論が得られた。これらの加速法の長所を生かしつつうまく組み合わせることにより, 既存のコードの収束性を大幅に改良することが可能である。

Convergence Acceleration Algorithms of the Iterative
Methods used in Nuclear Reactor Analysis Codes

Yasuaki Nakahara, Toichiro Fujimura

Division of Reactor Engineering, Tokai, JAERI

(Received February 4, 1974)

A systematic review has been made of the convergence acceleration algorithms for the iterative methods to solve the finite difference equations in generally used neutron transport S_N and diffusion codes. It also included the recent studies on new acceleration algorithms not tested yet for the nuclear codes. The acceleration algorithms were evaluated through the available results of numerical experiments.

It was found that the most promising are the coarse mesh rebalancing, the CCP, the line overrelaxation, the Chebyshev acceleration and the modified Wielandt methods. It is possible to make an extensive improvement in the convergence rates of codes by revising them, using a suitable combination of the methods mentioned.

目 次

1	序 論	1
2	核計算コードにおける反復解法の加速法	4
2.1	内側反復と外側反復	4
2.2	再釣合い法	5
2.3	緩和法	8
2.4	線優緩和法	12
2.5	中性子束外挿法	13
2.6	ウィーラント法	14
2.7	チェビシェフ加速法	16
3	数値実験による加速法の評価	20
3.1	再釣合い法	20
3.2	緩和法	21
3.3	線優緩和法	22
3.4	ウィーラント法	22
3.5	チェビシェフ加速法	22
4	核計算コードにおける加速法の改良	29
4.1	1次元 S_N コードの改良	29
4.2	2次元 S_N コードの改良	31
4.3	2次元拡散コードの改良	33
5	ま と め	45
	文 献	46

Contents

- 1 Introduction
 - 2 Acceleration Algorithms of Iterative Methods
 - 2.1 Inner and Outer Iterations
 - 2.2 Rebalancing Method
 - 2.3 Relaxation Method
 - 2.4 Line Overrelaxation Method
 - 2.5 Neutron Flux Extrapolation
 - 2.6 Wielandt Method
 - 2.7 Chebyshev Acceleration Method
 - 3 Evaluations of Acceleration Algorithms by Numerical Experiments
 - 3.1 Rebalancing Method
 - 3.2 Relaxation Method
 - 3.3 Line Overrelaxation Method
 - 3.4 Wielandt Method
 - 3.5 Chebyshev Acceleration Method
 - 4 Improvements of Acceleration Algorithms in Nuclear Analysis Codes
 - 4.1 Improvement of One-Dimensional S_N Codes
 - 4.2 Improvement of Two-Dimensional S_N Codes
 - 4.3 Improvement of Two-Dimensional Diffusion Codes
 - 5 Conclusions
- References

1. 序 論

原子炉中の中性子分布は一般に中性子輸送方程式あるいは中性子拡散方程式で記述されるが、これらの方程式は極めて単純な場合にのみ解析的に解けるに過ぎないので、一般の原子炉計算では反復解法が用いられている。単純な反復解法では収束性が悪いため、種々の加速法が工夫されており^{(1),(2)}、コードによってそれぞれ特徴ある用い方がされているが、どの加速法が最適かはそれほど明らかにはされていない。一方では、電子計算機の高速化、高容量化に伴って、加速法は無意味になりつつあるという誤った認識も一部にある。経済的観点からは計算機の進歩により演算がスピードアップされたとしても、加速法によって更に計算時間を短縮できればそれに越したことはない。また、安全解析や運転監視コードシステムにおいては、その目的上ますます高速化された計算手法の開発が必要であり、加速法の研究の重要性は将来ますます大きくなりこそすれ、小さくなるものではないであろう。確かに、現在用いられている加速法のうち将来その有効性が薄くなるものもあるかも知れないが、単にハードウェアの進歩だけでは解決できない問題もある。たとえば、拡散コードにおけるボイドや強い吸収のある領域の取扱いである。このような体系に対しては単純な反復法では収束解は得られず、このような場合でも精度のよい収束解を与える方法として現在知られているのは、粗メッシュ再釣合い法^{(2)~(7)}のみで、この方法は加速性能も非常に良いことが知られている⁽⁸⁾。

有限差分方程式 (finite difference equation) の型にした輸送方程式及び拡散方程式は一般に内側反復 (inner iteration) と外側反復 (outer iteration) を組合せた方法で解かれる。輸送方程式を解くには S_N 法⁽⁹⁾ が広く用いられており、 S_N コードで採用されている加速法には次のものがある。なお、加速法には内側反復、外側反復両方に有効なものもあり、必ずしもはっきり分類できないが、一応便宜的に次のように整理しておく。

(1) 内側反復の加速法

粗メッシュ再釣合い (coarse mesh rebalancing)

全体系再釣合い (whole system rebalancing)

各点における再釣合い (pointwise rebalancing)

中性子束外挿 (flux extrapolation)

(2) 外側反復の加速法

一因子上方散乱再釣合い (one factor upscatter rebalancing)

多群上方散乱再釣合い (multigroup upscatter rebalancing)

粗メッシュ上方散乱再釣合い (coarse mesh upscatter rebalancing)

チェビシェフ加速 (Chebyshev acceleration)

粗メッシュ固有値再釣合い (coarse mesh eigen value rebalancing)

核分裂中性子源の外挿 (fission source extrapolation)

拡散コードでは次の加速法が用いられている。

(1) 内側反復の加速法

粗メッシュ再釣合い

全体系再釣合い

中性子束外挿

多群再釣合い (multigroup rebalance)

線緩和 (line relaxation)

指数関数型 β 優緩和 (exponential- β overrelaxation)^(*)

逆行列型 β 優緩和 (inverse matrix- β overrelaxation)^(*)

単純優緩和 (simple overrelaxation)

DALOR (Alternating Direction Line Overrelaxation)

(2) 外側反復の加速法

中性子源の外挿

チェビシェフ加速

ウィーランドト法 (Wielandt method)

これらの加速法のどれを採用しているかはコードによってまちまちである。どのコードにはどんな加速法が用いられているかは、数値研の別の研究報告書にまとめてある。個々の加速法の具体的内容の説明は第2章で行うが、ここではそれらの特徴について簡単に議論しておく。

粗メッシュ再釣合い法が用いられている既存のコードは2次元 S_N コードの TWOTRAN 及び2次元拡散コードの GAMBLE-5 の二つである。TWOTRANでは空間についてのみ粗メッシュ化 (coarsing) を行っているが、GAMBLE-5ではさらにエネルギーメッシュについても行っている。GAMBLE-4⁽¹⁰⁾ や EXTERMINATOR-II⁽¹¹⁾ ではこの方法は用いられていないことに注意されたい。2次元拡散コードで普通に用いられている線優緩和法では正しい収束解が得られないのは次の3つの場合である。

1. 隣合う空間メッシュのカップリングに比較してエネルギー群間のカップリングが非常に弱い場合。
2. 空間メッシュ幅が非常に不揃いである場合。
3. 物質の性質が急激に変化する場合 (たとえば、ポイドのように拡散係数が極端に大きい領域がある場合)

粗メッシュ再釣合い法は、局所的な中性子分布の釣合せを行うのでこれらの場合にも極めて有効である。上記の三つの場合に収束性が乱れるのは、2次元拡散方程式を5点階差法で離散化した差分方程式の構造によるもので、1次元拡散コードではこの問題は生じない。ポイドのある体系をそのまま扱える2次元拡散コードは現在のところ GAMBLE-5のみということになる。

S_N コードでは1次元、2次元コード共にエネルギーメッシュについての粗メッシュ再釣合いは行われていない。 S_N コードでは一般に計算に時間がかかるので、多群計算は困難であるが、1次元 S_N コードである ANISN⁽¹²⁾ や DTF-IV⁽¹³⁾ 及び2次元 S_N コードの DOT⁽¹⁴⁾ や TWOTRANでエネルギーメッシュの粗メッシュ再釣合いを行うように改良すれば、計算時間を大巾に短縮でき、 S_N コードによる多群計算が実用的になり、高速炉設計や遮へい計算への S_N 法の応用が容易になると期待される。我々の研究室で現在この作業を進

(*) これは筆者が便宜的につけた名称で、正式の呼称ではない。

めているところである。

FroehlichのGAMBLE-5を用いた計算例では、粗メッシュ再釣合い法を適用すると計算時間が1/4になったと報告されている⁽³⁾。

ADI (Alternating Direction Implicit) 法⁽¹⁵⁾ (先にリストアップしたDALOR法もこの一種)は効力の大きい加速法であるが、多次元多群問題ではデータ処理が難しいので、2次元コードでは一般に用いられておらず、CITATION⁽¹⁶⁾だけがDALORを組み込んでいるに過ぎない。

その他の内側反復加速法については第2章で議論する。次に外側反復加速法について簡単に議論しておく。

チェビシェフ加速は1次元計算ではかなり有効である。2次元問題では、理由ははっきりしないが、効果が落ちる。Reedの行った1次元の均質平板の S_N 計算では、問題にもよるが、反復回数はチェビシェフ加速を用いると1/5~1/2になる⁽¹⁷⁾。チェビシェフ加速は原子炉が臨界に近い時が一番有効で、未臨界になるほど効力が悪くなる。その理由については第2章で議論する。この加速法はDTF-IVとANISNでは用いられていない。DTF-IVの改訂版であるIDFX⁽¹⁸⁾及び γ 線用DTF版のDTF-69⁽¹⁹⁾ではチェビシェフ加速を組み込んでいるが、これらのコードは未だ原研では入手されていない。

拡散計算におけるチェビシェフ加速については、1群モデルで熱出力300MWの炉についてFroehlichがチェックしている⁽³⁾。その結果によると、この加速法を用いた場合、収束条件0.001の時、単純な外側反復の場合と比較して反復回数は、高温ガス炉、沸騰水型軽水炉、加圧水型軽水炉、ガス冷却高速炉、液体金属冷却高速増殖炉に対して、各々、4%、1.4%、1.3%、7.2%、6.8%であり、この方法はかなり効果的であることが分る。

ウィーラント法を反復解法に用いた場合、その収束性は固有値の初期見積り値に強く依存し、これが低すぎると、解の正值性がこわれ、収束しなくなることがある⁽²⁰⁾。一方、見積りが高すぎると、収束が悪くなる。見積り値が高い方から固有値の真の値に近づくにつれ、収束性は一般に良くなるが、非常に近くなった場合かえって収束が悪くなる例も報告されており⁽²⁾、固有値の初期見積りの難しさが、ウィーラント法の欠点となっている。ウィーラント法を採用しているのはGAMBLE-5のみである。

以上この章で議論しなかった問題と数値実験例については、第2章と第3章で、また既成のコードで未だ用いられていない加速法については、その応用の可能性も含めて、第4章で議論する。

2. 核計算コードにおける反復解法の加速法

2.1 内側反復と外側反復

中性子輸送方程式および拡散方程式を簡単に次のように書こう。

$$L\psi = S\psi + q, \quad (1)$$

ここで輸送方程式の場合は，エネルギー群インデックスを g として，

$$[L\psi(r, \Omega)]_g = \Omega \cdot \nabla \psi_g(r, \Omega) + \Sigma_{t, g}(r) \psi_g(r, \Omega), \quad (2)$$

$$[S\psi(r, \Omega)]_g = \sum_{g'} \int_{\Omega'} d\Omega' \Sigma_{g' \rightarrow g}(r, \Omega' \rightarrow \Omega) \psi_{g'}(r, \Omega'), \quad (3)$$

であり，拡散方程式の場合は

$$[L\psi(r)]_g = -D_g(r) \nabla^2 \psi_g(r) + \Sigma_{t, g}(r) \psi_g(r), \quad (4)$$

$$[S\psi(r)]_g = \sum_{g'} \Sigma_{g' \rightarrow g}(r) \psi_{g'}(r), \quad (5)$$

で与えられる。 q は外部中性子源のない時は核分裂中性子源の項で， $\psi_g(r, \Omega)$ または $\psi_g(r)$ に依存する。散乱行列 S を次のように分割する。

$$S = S_d + S_s + S_u, \quad (6)$$

ここで S_d , S_s , S_u は各々下方散乱，群内散乱 (ingroup scattering) 及び上方散乱に対応する。そこで反復を

$$(L S_d - S_s) \psi^{(j+1)} = S_u \psi^{(j)} + q^{(j)} \quad (7)$$

のアルゴリズムで行うのを外側反復 (outer iteration) または中性子源反復 (source iteration) と呼ぶ。(7)式の左辺に S_s 項があるため，(7)を行列逆転で直接解くは困難なので，さらに内側反復 (inner iteration) または中性子束反復 (flux iteration) と呼ばれるアルゴリズムを用いる。エネルギーの g 群への中性子源は

$$q_g^{(j+1)} = (S_d \psi^{(j+1)})_g + (S_u \psi^{(j)})_g + q_g^{(j)} \quad (8)$$

で与えられ，

$$L_g \psi_g^{(j+1), (k+1)} = S_{s g} \psi_g^{(j+1), (k)} + q_g^{(j+1)} \quad (9)$$

のアルゴリズムを内側反復と呼ぶ。ここで k は内側反復の回数である。

(7)式と(9)式の計算を単純に繰返したのでは，収束が悪いため，種々の加速法が工夫されている。

2.2 再釣合い法 (Rebalance method)

第1章で述べたように、再釣合い法にはいくつかの方法があるが、最も一般的なものは粗メッシュ再釣合い法で、他の方法はその特別な場合として説明できる。再釣合い法というのは、一口で表現すると、近似収束解に正の数をかけて、scaling によって注目する各々の反復計算中の中性子分布の釣合いをはかり、それによって近似度を上げようというものである。この補正因子 (scaling factor または rebalance factor) の決め方により種々の再釣合い法が与えられる。そこで先ず粗メッシュ再釣合い法から議論を進めることにする。

(1) 粗メッシュ再釣合い法

この方法を最初に提唱したのは Wachspress⁽²¹⁾ で、後に多群拡散計算へは Froehlich により拡張され⁽³⁾、また輸送方程式への応用は Nakamura によって行われた⁽⁴⁾。Froehlich のアルゴリズムは GAMBLE-5 で、Nakamura のアルゴリズムは TWOTRAN で各々用いられている。

実際にこの方法を用いている GAMBLE-5 と TWOTRAN に即してまず詳細メッシュの切り方を見てみよう。GAMBLE-5 では Fig.1 のように両者の分割点は一致しないで、粗メッシュ点は詳細メッシュ点の中間に入り、入力データとしては両者の各方向の全メッシュ点数を各々与える。TWOTRAN では Fig.2 に見るように、両者は重なり合い、粗メッシュ番号と各粗メッシュ内の詳細メッシュの数を入力する。

詳細メッシュ (s, t, g) 上の5点差分拡散方程式は次のように書ける。ここで s と t は空間メッシュ、 g はエネルギー群のインデックスである。

$$\begin{aligned} \alpha_0(s, t, g) \phi_g(s, t) &= \alpha_1(s, t, g) \phi_g(s, t-1) + \alpha_2(s, t, g) \phi_g(s-1, t) \\ &+ \alpha_3(s, t, g) \phi_g(s, t+1) + \alpha_4(s, t, g) \phi_g(s+1, t) \\ &+ \frac{\chi_g}{\lambda} \left\{ \sum_{j=1}^G \beta_j(s, t) \phi_j(s, t) \right\} + \sum_{j \neq g}^G \gamma_{j \rightarrow g}(s, t) \phi_j(s, t) \end{aligned} \quad (10)$$

ここで、係数 $\alpha_0 \sim \alpha_4$ は拡散係数を含む量であるが、その具体的表現については、GAMBLE-5 のマニュアル⁽⁶⁾を参照されたい。 χ_g は核分裂中性子のスペクトル、 λ は固有値、 β_j は $\nu \Sigma_f(s, t)_j$ 、 γ は散乱行列である。粗メッシュブロック $(\tilde{s}, \tilde{t}, \tilde{g})$ 内に詳細メッシュが $s = s_1 \sim s_2$, $t = t_1 \sim t_2$, $g = g_1 \sim g_2$ 含まれているとしよう。この粗メッシュブロックに対する中性子束補正因子 f は (10) 式と同じ形の次の式を満足する。

$$\begin{aligned} \alpha_0(\tilde{s}, \tilde{t}, \tilde{g}) f(\tilde{s}, \tilde{t}, \tilde{g}) &= \alpha_1(\tilde{s}, \tilde{t}, \tilde{g}) f(\tilde{s}, \tilde{t}-1, \tilde{g}) \\ &+ \alpha_2(\tilde{s}, \tilde{t}, \tilde{g}) f(\tilde{s}-1, \tilde{t}, \tilde{g}) + \alpha_3(\tilde{s}, \tilde{t}, \tilde{g}) f(\tilde{s}, \tilde{t}+1, \tilde{g}) \\ &+ \alpha_4(\tilde{s}, \tilde{t}, \tilde{g}) f(\tilde{s}+1, \tilde{t}, \tilde{g}) + \frac{\chi(\tilde{g})}{\lambda} \left\{ \sum_{i=1}^{\tilde{G}} \beta_i(\tilde{s}, \tilde{t}) f(\tilde{s}, \tilde{t}, i) \right\} \\ &+ \sum_{i \neq \tilde{g}}^{\tilde{G}} \tilde{\gamma}_{i \rightarrow \tilde{g}}(\tilde{s}, \tilde{t}) f(\tilde{s}, \tilde{t}, i). \end{aligned} \quad (11)$$

行列形式で書くと、

$$A f = \frac{1}{\lambda} B f . \quad (12)$$

係数 $a_0 \sim a_4$ 等は粗メッシュブロック内の詳細メッシュ点の中性子束から計算される。その具体的表現については GAMBLe-5 のマニュアルを参照されたい。

GAMBLe-5 では (12) 式を解くのに、ウィーラント法を用いる^(*)。したがって、議論は後のウィーラント法の節で行うことにする。

TWO TRAN では中性子の角度分布 (angular flux) を反復毎にストアすることはせず、内側反復は次式により行う。

$$\begin{aligned} & A_{i+\frac{1}{2}} I_{i+\frac{1}{2}} - A_{i-\frac{1}{2}} I_{i-\frac{1}{2}} B_{j+\frac{1}{2}} J_{j+\frac{1}{2}} - B_{j-\frac{1}{2}} J_{j-\frac{1}{2}} \\ & + \Sigma_t V \phi = V \Sigma_{g \rightarrow g} \phi^P + V Q ; \end{aligned} \quad (13)$$

ここで、 A, B は各々 i 方向、 j 方向のセル表面積、 V はセル体積、 ϕ は中性子束、 ϕ^P は前回の反復計算で得られた中性子束、 Q は中性子源である。インデックスの付け方については、Fig. 3 を参考にされたい。 I と J は次のように角度について角度分布 N を積分したものである。

$$\left. \begin{aligned} I_{i \pm \frac{1}{2}, j} &= \sum_m w_m \mu_m N_{i \pm \frac{1}{2}, j, m} \\ J_{i, j \pm \frac{1}{2}} &= \sum_m w_m \eta_m N_{i, j \pm \frac{1}{2}, m} \end{aligned} \right\} \quad (14)$$

ここで、 m は角度メッシュのインデックス、 w_m は角度積分の重み、 μ と η は、各々、 i, j 方向余弦である。TWO TRAN でも、(12) 式の場合と同じように、粗メッシュブロックについて、(吸収+洩れ) = (源) となるように補正因子を決めるのであるが、手順は GAMBLe の場合とは異なる。TWO TRAN では、エネルギーについては粗メッシュ再釣合いをさせない。粗メッシュブロックを (k, l) で表わすと、まず、このブロック境界での上方、下方、左方向の流れを計算する。すなわち、

$$F U_{k, l \pm \frac{1}{2}} = \sum_{i \in k} \sum_{\eta_m > 0} w_m \eta_m B_{i, l \pm \frac{1}{2}} N_{i, l \pm \frac{1}{2}, m} \quad (15a)$$

$F D_{k, l \pm \frac{1}{2}}$ は (15a) 式で $\eta_m < 0$ について和をとる。

$$F R_{k \pm \frac{1}{2}, l} = \sum_{j \in l} \sum_{\mu_m > 0} w_m \mu_m A_{k \pm \frac{1}{2}, j} N_{k \pm \frac{1}{2}, j, m}, \quad (15b)$$

$F L_{k \pm \frac{1}{2}, l}$ は (15b) 式で $\mu_m < 0$ について和をとる。さらに粗メッシュブロック内で

(*) 詳細メッシュ上での反復計算では、ウィーラント法は用いず、線優緩和法を用いる。

の吸収及び中性子源を各々 AB , QQ とすると, 粗メッシュブロックについての再釣合いの式は, 補正因子を f とじて, 次のようになる。

$$\begin{aligned}
 & f_{kl} \left(FR_{k+\frac{1}{2}} + FL_{k-\frac{1}{2}} + FU_{l+\frac{1}{2}} + FD_{l-\frac{1}{2}} + AB \right) \\
 & = QQ + f_{k+1, l} FL_{k+\frac{1}{2}} + f_{k-1, l} FR_{k-\frac{1}{2}} \\
 & \quad + f_{k, l+1} FD_{l+\frac{1}{2}} + f_{k, l-1} FU_{l-\frac{1}{2}} \quad (16)
 \end{aligned}$$

(16) 式は変分法によっても導くことができる⁽⁴⁾。(16) 式を反復解法で解いて f_{kl} が求められる。

当然のことながら, 粗メッシュブロックに含まれる詳細メッシュの数があまり多いと, 加速性は悪くなる。

(2) 全体系再釣合い法

これは全体系で釣合いのとれるように一個の補正因子で中性子束を補正する方法である。全体系についての洩れ (NL), 吸収 (AB), 源 (Q) を合せて,

$$f = \frac{\sum_{k,l} QQ_{kl}}{NL + \sum_{k,l} AB_{kl}} \quad (17)$$

により, 補正因子 f を決定する。これは粗メッシュ再釣合い法で一つの粗メッシュに詳細メッシュの全てが含まれる特別の場合で, (16) 式よりこの方法の方がはるかに簡単であるが, 加速性能は劣る。この方法は収束の最終段階で行うのが一般的である。

(3) 各点における再釣合い法

これは名称の通り (12) や (16) 式のように粗メッシュをとらず, 詳細メッシュについて再釣合いを行う方法である。この方法は, 2次元計算では面倒でメリットが少ないため, 主として 1次元 S_N コードで用いられている。

再釣合い法は外側反復の加速にも用いられる。次に外側反復の加速法についてのまとめを行う。

(4) 粗メッシュ固有値再釣合い法及び粗メッシュ上方散乱再釣合い法

これは(1)の粗メッシュ再釣合い法の変形で, エネルギー全群について縮約してから, (16) 式を解いて f_{kl} を求めるのが粗メッシュ再釣合い法であるが, 粗メッシュ固有値再釣合い法では, (16) 式の源項 QQ を $f_{kl} FT_{kl}/k_{eff}$ とする。ここで FT_{kl} は空間の粗メッシュブロック内の全核分裂中性子源である。この加速法は上方散乱がある場合にも有効で, これを粗メッシュ上方散乱再釣合い法と呼び, 外側反復の初期の段階で用いられる。

(5) 一因子上方散乱再釣合い法

これは上方散乱がある場合の外側反復の加速法であり, $DTF-IV$, $ANISN$, DOT で用いられている。中性子源については, あるエネルギー群内で発生する源, その群へ減速によって入って来る源は外側反復の際逐一更新されるが, 下の群から上方散乱によって入ってく

る源は更新されないため、外側反復の収束が悪くなる。そのため上方散乱による源の項に次の全体系全エネルギー群について縮約した補正因子を用いる。

$$f^{up} = \frac{\sum_{g,i,k} \frac{x_g}{k_{eff}} FT_{ik}}{\sum_{g,i,k} \frac{x_g}{k_{eff}} FT_{ik} - \sum_{g,i,k} \Sigma_g^{up} (\phi_{gik}^{new} - \phi_{gik}^{old}) V_{ik}} \quad (18)$$

(18) 式で用いられている記号は、既に出てきているものと同じ意味を持っている。

多群上方散乱再釣合い法は I D E X (18) で用いられており、他のコードでこれを用いているものはない。文献(18)は原研に入っていないので、具体的な形は不明であるが、(18)式でエネルギーについての縮約を行わず、 f^{up} にエネルギー依存性を持たせるものと推定される。

2.3 緩和法 (Relaxation method)

緩和法は行列型方程式の反復解法の加速に用いられるもので、核計算コードでは主として拡散コードで用いられている。次の行列方程式を反復解法で解くことを考えよう (22), (23)。

$$A X = k \quad (19)$$

ここで A は $N \times N$ の実正則行列、 k は列ベクトルで既知量であるとする。ここで A を次のように分解する。

$$A = D - L - U, \quad (20)$$

$$D = \begin{bmatrix} A_{1,1} & & & \\ & A_{2,2} & & 0 \\ & & & \\ & & & \\ 0 & & & A_{N,N} \end{bmatrix}, \quad L = \begin{bmatrix} 0 & & & & \\ A_{2,1} & 0 & & & \\ \vdots & & \ddots & & \\ A_{N,1} & & & A_{N,N-1} & 0 \end{bmatrix}$$

$$U = \begin{bmatrix} 0 & A_{1,2} & & A_{1,N} \\ & & & \\ & & & \\ 0 & & & A_{N-1,N} \\ & & & & 0 \end{bmatrix}, \quad (21)$$

ここで、対角部分行列 $A_{j,j}$ は正則 (non-singular) であるとする。そうすると、(19) の反復解法のアルゴリズムとして単純には

$$DX^{(q+1)} = (L+U)X^{(q)} + R \quad (22)$$

が考えられるが、これの収束性を良くするために、次のアルゴリズムを定義する。

$$DX^{(q+1)} = \beta \{ LX^{(q+1)} + UX^{(q)} + k \} + (1-\beta)DX^{(q)} \quad (23)$$

ここで、 β が加速因子で、 $\beta = 1$ の場合を Gauss-Seidel, Liebmann の方法と呼び、 $0 < \beta < 1$, $1 < \beta < 2$ の場合を各々連続劣緩和法 (successive underrelaxation method) 及び連続優緩和法 (successive overrelaxation method = SOR) と呼んでいる。

拡散コードではいずれも優緩和法を用いている。加速因子の最適詳細によれば、 β の最適値は、反復行列 $J = D^{-1}(L+U)$ のスペクトル半径を ρ とすると、

$$\beta_b = \frac{2}{1 + \sqrt{1 - \rho^2}} \quad (24)$$

で与えられる(1),(2),(23)。 $0 < \rho < 1$ であるから、 $1 < \beta_b < 2$ であり、優緩和法を拡散コードで採用していることは妥当であることが分る。スペクトル半径を事前に決めるのは、一般的には困難なので、加速因子の最適値を決めるのにも、経験的要素が入ってくるのはやむを得ないことかも知れない。Hagemann の数値実験例によると、加速性は β の値にかなり敏感であり、また、最適値を用いると、(22) 式の加速をしない単純な場合に比較して、解く問題にもよるが、反復回数が 7% ~ 0.08% になることが報告されている(23)。 β の値が最適値から極くわずか ($\beta - \beta_b = 0.1756$) ずれただけで、反復回数が最適反復数の 6.3 倍にもなる例も報告されている(23)。加速法に関する種々の数値実験例については第 3 章でまとめて議論する。

(1) 逆行列型 β 優緩和法 (Inverse matrix- β overrelaxation)

この呼称は筆者が仮につけたものであるが、この加速法は(23)式の方法の変形である。核計算での固有値問題は行列表示で

$$A\phi = \lambda M\phi \quad (25)$$

と書ける。行列 A を適当に分割して、次のような反復アルゴリズムを採用する。加速因子を β とし、

$$\phi^{(n+1)} = \beta (V - \beta L - \beta \lambda_n W)^{-1} (\lambda_n M - A) \phi^{(n)} + \phi^{(n)}. \quad (26)^*$$

このアルゴリズムは EXTERMINATOR (24) で採用されているが、 ϕ^n から $\phi^{(n+1)}$ への変化率が 0.1% になってから使いようになっている。変化率がこれより大きい間は、次項で述べる指数関数型 β 優緩和法を用いる。

先に述べたように加速性は加速因子 β の値に大きく作用される。スペクトル半径 $\rho(A)$ は空間メッシュ、エネルギー群の分け方に依存するので、 β の最適値を事前に決めるのが難しいため、EXTERMINATOR は次のような方法を用いている。

まず、 $\beta = 1.9$ で計算を始め、そのまま 20 回反復計算を行い、10 回目の時との固有ベクトル (中性子束) の収束率 $\epsilon_n = (\phi^{(n)} - \phi^{(n-1)}) / \phi^{(n-1)}$ の比 $|\epsilon_{10} / \epsilon_{20}|$ を求め、もしこれが 1.5 より大きい場合は、さらに $\beta = 1.9$ とし 20 回反復計算を行う

という手順を繰返す。この比が1.5以下になった時は、さらに、

$$\left| \frac{\varepsilon_n}{\varepsilon_{n-1}} (1 + |\varepsilon_{n-1}|) \right|$$

の値を求め、 $n = 10 \sim 19$ の間にこの比が1.0より大きい場合が3回以上あった時は、 β の値を0.1だけ小さくする。この条件が満たされない場合は2.5節で述べる中性子束の外挿法を用いる。このアルゴリズムは経験的なもので、数学的論拠によるものではない。

加速因子の値の選び方にはCarre'の方法、Reidの方法及び α 、-RQ方法などがある。これらの方法については第4章で議論する。

(2) 単純優緩和法及び指数関数型優緩和法 (Simple overrelaxation method and Exponential- β overrelaxation method)

拡散コードで用いられているのは非常に単純なもので、EXTERMINATORで用いられている単純優緩和法は

$$\phi^{(n+1)} = \phi^{(m)} + \beta (\phi^{(n+1)*} - \phi^{(m)}) \quad (26)$$

で与えられる。ここで $\phi^{(n+1)*}$ は(26)式を適用する以前に、他の手段(線緩和法等、2.4節をみよ)で求めた値である。この方法では負の中性子束が発生する可能性があるため、それを防止するためにEXTERMINATORでは次のような手段を用いている(27)。

(26)式を次のように変形する。

$$\phi^{(n+1)} = \phi^{(n)} \left[1 + \beta \left(\frac{\phi^{(n+1)*}}{\phi^{(n)}} - 1 \right) \right]. \quad (27)$$

(27)式で、 $\phi^{(n+1)*} / \phi^{(n)} - 1$ が負の時は、 $\phi^{(n+1)}$ が負になる可能性がある。その場合は $1 - x \approx e^{-x}$ の関係を用いて、

$$\phi^{(n+1)} = \phi^{(n)} \exp \left[-\beta \left| \frac{\phi^{(n+1)*}}{\phi^{(n)}} - 1 \right| \right] \quad (28)$$

とする。(26)式は線型外挿の式であるが、逆に $\phi^{(n+1)*} / \phi^{(n)} - 1$ が正でも、値が大きい場合は、過外挿(excessive extrapolation)になるため、

$$\phi^{(n+1)} = \phi^{(n)} \left\{ 2 - \exp \left[-\beta \left(\frac{\phi^{(n+1)*}}{\phi^{(n)}} - 1 \right) \right] \right\} \quad (29)$$

を用いる。このようにすると、(28)と(29)式から $\phi^{(n)} \exp(-\beta) \leq \phi^{(n+1)} \leq 2\phi^{(n)}$ となるので、過外挿を防止できる。(28)、(29)式で与えられる加速法を指数関数型優緩和法と呼ぶ。

CITATION⁽¹⁶⁾でも(26)式を用いるが、中性子束が負になることと過外挿を防止

(*) 行列V, L, Wの定義はEXTERMINATORのマニュアルには書かれていない。

する手段は、EXTERMINATORとは異なり、次のような束縛条件を用いている。

$$0.5 \phi^{(n+1)*} < \phi^{(n+1)} < \phi^{(n+1)*} + \phi^{(n)}. \quad (30)$$

CITATION 方式では、指数関数の計算をしなくてよいので、EXTERMINATOR 方式より計算能率はよい。

加速因子 β の値は EXTERMINATOR では第(1)項で説明した方法を用いるが、CITATION では次のようにする。まず(22)式のヤコービ行列 $J \equiv D^{-1} (L+U)$ のスペクトル半径をコードの中で計算する。何度も述べてきたように、正確なスペクトル半径の値を事前に決めることはできないので、

$$\rho(J) = \frac{1}{N} \sum_{i=1}^N \cos \left(\frac{\pi}{I_i + 1} \right) \quad (31)$$

で評価する。ここで N は空間の次元数、 I_i は i 方向の空間メッシュの数である。(31)式の値から β の初期値を(24)式、すなわち

$$\beta = \frac{2}{1 + \sqrt{1 - \rho^2}}$$

として決める。この値で反復を行っても、収束が良くない場合には

$$\beta^{(n)} = \frac{6 \beta^{(n-1)} + 1}{7} \quad (32)$$

によって値を下げる。一方、

$$r^{(n)} = \phi^{(n)} / \phi^{(n-1)} \quad (33)$$

$$\mu^{(n)} = \left(\frac{r^{(n)} - 1}{r^{(n-1)} - 1} \right) r^{(n-1)} \quad (34)^{(*)}$$

を求め、 $\mu^{(n)} > 1$ 、 $r^{(n)} > 2$ の時は

$$\beta^{(n)} = \frac{4 \beta^{(n-1)} + 1}{5} \quad (35)$$

とする。このようにして β の値を小さくして行くが、劣緩和法をさけるため $\beta \geq 1$ とする。

このような β の低減法が効果を発揮するには、 β の初期値、EXTERMINATOR では 1.9、CITATION では(31)式が、最適値より大きいことが必要である。

この加速法を 2.4 節の線緩和法と組み合わせる時は、スペクトル半径は、(31)式の $\rho(J)$ から

$$\rho = \frac{\rho(J)}{2 - \rho(J)} \quad (36)$$

(*) 文献(27)の Eq. (730-11)は誤りである。

により決める(28)。(31)式及び(36)式から求めた β は最適値より大きい値を与えるという証明がされているかどうかは不明である。

一次元拡散計算では本来計算時間が短かくてすむので、加速はあまり重要視されていない。WANDA⁽²⁴⁾系列のコード(25),(26)で用いられている唯一の加速法は(27)式に類似の中性子源についての線型外挿である。中性子束 ϕ_g から中性子源は

$$G = \sum_g (\nu \Sigma_f)_g \phi_g \quad (37)$$

により計算され、固有値 λ は

$$\lambda = \int G(\vec{r}) d\vec{r} \quad (38)$$

で定義される。次に λ で規格化した新しい中性子源

$$S = G / \lambda \quad (39)$$

を求める。(39)式により求めた源を $S^{(n)*}$ とすると、加速法は

$$S^{(n)} = S^{(n)*} + \beta (S^{(n)*} - S^{(n-1)}) \quad (40)$$

によって定義され、これにより新しい源を定める。

(40)式で用いられる β の値はWANDA, EXPANDA⁽²⁵⁾では入力するようになっている。GURNETではコードの中で0.8に固定されている。いずれにしても、2次元コードの場合のように、反復過程で β の値を修正することはしない。(40)式で用いる β の最適評価は未だ行われていないようである。

2.4 線優緩和法 (Line overrelaxation method)

この方法は、緩和法という名前がついているが、前節の加速因子を用いるものではなく、二次元計算において、特定の方向のメッシュライン上で固有ベクトル(中性子束)の緩和(relaxation)を行うことによって計算速度を速めようというものである。ここでいう緩和という言葉の意味は、指定された特定の方向についての反復計算を行い、そのスイープ線上にないメッシュ点に対する項の値は源の項に含めてupdateすることである。たとえば、二次元拡散コードで用いられる5点有限差分方程式(10)式をs方向に行緩和(relaxation along columns)させるとすると、(10)式は次のように書ける。

$$\begin{aligned} & \alpha_4 (s+1, t, g) \phi_g(s+1, t) + \alpha_2 (s-1, t, g) \phi_g(s-1, t) \\ & - \alpha_0 (s, t, g) \phi_g(s, t) + C(s, t, g) = 0, \end{aligned} \quad (41)$$

ここで $C(s, t, g)$ は $\phi_g(s, t \pm 1, \cdot)$, $\phi_g \neq g(s, t)$ のupdateされた値から求められる。(41)式はメッシュラインs上の3点差分方程式(10)式を直接解くよりも(41)式を解くことははるかに容易である。同じようにして列緩和(relaxation along rows)が定義できる。

(41) 式の係数は 3 軸対角行列 (tri-diagonal matrix) を形成するので、(41) 式の解を求めるには、この行列の逆転すればよいわけであるが、EXTERMINATOR では逆行列による直接解法を避けて、(41) 式を漸化式の型：

$$\phi_g(s, t) = \frac{mg(s, t) + \phi_g(s+1, t)}{q_g(s, t)} \quad (42)$$

に変形して計算している。行列が大ききものでない限り、3 軸対角行列の逆転は容易なので、計算精度と計算速度の点で直接法と漸化法の得失を評価してみる価値はあるであろう。

線緩和を行う方向は入力データで指定する。領域毎の群定数の値の変化が小さい方の方向を選んだ方が、収束及び精度共に良い。

ADI (Alternating Direction Implicit Iteration) 法というのは、この線緩和を行う方向を交互に変更するものであるが⁽²⁹⁾、一般性に欠けるといふことと、多群多領域系に対してはプログラム作成及びデータ処理が難しいということから 2 次元コードではあまり用いられていない。しかし、ADI 法の改良及び応用に関する研究は動特性の分野で盛んに行われている。動特性コードで用いられている計算手法については、本報とは別の当研究室の研究報告書にまとめられるので、ここでは詳述せず、若干の文献を挙げておくに止める^{(29), (30), (31), (32), (33)}。

ADI 法を用いているのは CITATION であるが⁽³⁴⁾、単純な ADI 法ではなく、前節の (26) と (30) 式で定義された優緩和法と組合せた DALOR (Alternating Direction line Overrelation) 法を上方散乱のない問題に対してのみ適用するようになっている。この方法を用いた場合の収束率は問題によってかなり左右され、必ずしも内側反復の加速に効果的とは限らないようである⁽³⁴⁾。

2.5 中性子束外挿法 (Flux extrapolation)

中性子束外挿は S_N コードと拡散コードでは本質的に異った用い方をされているので、両者の場合を区別して議論する必要がある。

S_N コードの場合には中性子束外挿は、 S_N アルゴリズムの中で中心的な役割を持っている手法の一つで、ダイヤモンド差分法と一般に呼ばれているものである。すなわち、

$$\left. \begin{aligned} N_{i+1, m} + N_{i, m} &= 2 N_{i+1/2, m}, \\ N_{i+1/2, m+1/2} + N_{i+1/2, m-1/2} &= 2 N_{i+1/2, m}, \end{aligned} \right\} \quad (43)$$

ここでの記号の意味については (14) 式の後の説明を参照されたい。この外挿方式のため、 S_N コードでは中性子束が負になることがあるが、この問題については別の機会に論ずる。

拡散コードでは中性子束外挿は補助的な役割りしか持っておらず、EXTERMINATOR では他の加速法が効果を発揮しない場合にのみ用いるようになっている。その判定には中性子束の収束率 ϵ_n の減衰率が 1.5 以下で、かつ

$$\left| \frac{\varepsilon_n}{\varepsilon_{n-1}} (1 + |\varepsilon_{n-1}|) \right| < 1.0$$

の条件が満たされた場合に中性子束外挿を行う。

EXTERMINATORとCITATIONで用いられている外挿法は基本的には同じなので、EXTERMINATORに沿って説明する。n回目の反復で求められた中性子束 $\phi^{(n)}$ と外挿中性子束 $\bar{\phi}$ は

$$\phi^{(n)} = \bar{\phi} + R \sigma^n \quad (44)$$

の関係にあると仮定する。外挿誤差項 $R \sigma^n$ のRは常数であるが、実際の計算には必要ない。(44)式と

$$\phi^{(n-1)} = \bar{\phi} + R \sigma^{n-1} \quad (44')$$

から、Rを消去すると、外挿中性子束は

$$\bar{\phi} = \phi^{(n)} + \frac{\sigma}{1-\sigma} (\phi^{(n)} - \phi^{(n-1)}) \quad (45)$$

で与えられ、これを外挿公式として用いる。減衰因子 σ は次のようにして計算する。(44)、(44')及び

$$\phi^{(n-2)} = \bar{\phi} + R \sigma^{n-2} \quad (44'')$$

から、Rと $\bar{\phi}$ を消去すると

$$\sigma = (\phi^{(n)} - \phi^{(n-1)}) / (\phi^{(n-1)} - \phi^{(n-2)})$$

となり、さらに

$$\varepsilon_n = [(\phi^{(n)} - \phi^{(n-1)}) / \phi^{(n-1)}]_{\max}$$

を用いると、

$$\sigma = (\varepsilon_n / \varepsilon_{n-1}) (1 + \varepsilon_{n-1}) \quad (46)$$

が得られる。

外挿によって中性子束が負になる場合にはこの方法は使えず、他の加速法を用いなければならない。

2.6 ウイーラント法(The Wielandt method)

これは外側反復解法の加速に用いられている方法の一つで、GAMBLE-5で粗メッシュ化(12)式の解法として採用されている⁽⁶⁾。まず、(12)式を

$$M f \equiv \left(A - \frac{1}{\lambda_e} B \right) f = \left(\frac{1}{\lambda} - \frac{1}{\lambda_e} \right) B f \quad (47)$$

と書くと次のアルゴリズムによって定義される。

a. 前方消去

$$\begin{aligned} H_1 &= -B_1^{-1} C_1, \\ H_i &= -(B_i + A_i H_{i-1})^{-1} C_i, \quad i=1, 2, \dots, I-1, \\ r_1 &= B_1^{-1} q_1, \\ r_n &= (B_n + A_n H_{n-1})^{-1} C_n, \quad n=2, 3, \dots, I. \end{aligned} \quad (53)$$

b. 後方代入

$$\begin{aligned} f_I &= r_I, \\ f_i &= r_i + H_i f_{i+1}, \quad i=I-1, I-2, \dots, 1, \end{aligned} \quad (54)$$

ここで行列 $(B_i + A_i H_{i-1})^{-1}$ は一回計算するだけで、それを保存しておけばよく、そのため計算時間が節約される。

ウィーラント法の収束性については次の定理が成り立つ⁽³⁵⁾。

定理. $0 < \lambda_e \leq \infty$ の時、全ての行列

$$(B_i + A_i H_{i-1}), \quad i=1, 2, \dots, I$$

が正則 (non-singular) でかつ非負逆行列を持っている場合にのみ

$$\lambda_1 < \lambda_e \leq \infty$$

が成り立ち、かつこの場合には、ウィーラント法による解は基本モードの固有値と固有ベクトルに収束する。

固有値の初期見積りの精度に対する厳しい要求をやわらげ、かつ、収束の加速性のよい方法として Feldmark が修正ウィーラント法を提案しているが⁽³⁶⁾、これについては第4章で議論する。

2.7 チェビシエフ加速法 (Chebyshev acceleration method)

我々のベンチマークテストで使用を予定してコードにはチェビシエフ加速法を用いているものはないが、一次元 SN コード DTF 系列の 1 DFX⁽¹⁸⁾ 及び DTF 69⁽¹⁹⁾ で用いているので、ここでこの方法について議論しておく。

$$A \phi = \frac{1}{\lambda} B \phi$$

与えられる行列方程式の反復アルゴリズムを

$$\begin{aligned} A \phi^{(n)} &= \psi^{(n-1)}, \\ S^{(n)} &= B \phi^{(n)}, \end{aligned}$$

$$\lambda^{(n)} = \frac{\langle s^{(n)} | s^{(n)} \rangle}{\langle s^{(n)} | \psi^{(n-1)} \rangle}, \quad (56)$$

$$\psi^{(n)} = \frac{1}{\lambda^{(n)}} s^{(n)}$$

で定義する。反復行列 T_p を

$$T_p = B A^{-1} \quad (57)$$

で定義すると、

$$\lambda^{(n)} \psi^{(n)} = T_p \psi^{(n-1)} \quad (58)$$

を得る。 T_p の固有値を λ_i とし、優越比 (dominance ratio) を λ_i と基本モードの固有値 λ_1 から、

$$\sigma_p = \max_{i \neq 1} |\lambda_i| / |\lambda_1| \quad (59)$$

で定義する。この σ_p が 1 に近い時には、外側反復アルゴリズム (58) 式の収束は非常に遅い。チェビシェフ加速法はこのような場合の加速に用いられるのであるが、チェビシェフ加速の効果は $\sigma_p \rightarrow 1$ につれて減少する。

(56) 式の源を次の外挿式で置きかえる (20)。

$$\begin{aligned} \psi^{(n)} = \psi^{(n-1)} + \alpha^{(n)} \left[\frac{1}{\lambda^{(n)}} s^{(n)} - \psi^{(n-1)} \right] \\ + \beta^{(n)} [\psi^{(n-1)} - \psi^{(n-2)}] \end{aligned} \quad (60)$$

(60) 式で、 $\psi^{(n)}$ はチェビシェフ多項式 T_n を用いて

$$\psi^{(n)} = \sum_i \frac{T_n(z_i)}{T_n(\frac{1}{\mu})} c_i v_i \quad (61)$$

と展開できるとして、(61) 式を (60) 式に代入し、チェビシェフ多項式の性質を利用して、(60) 式の係数 α と β の表現が求められる。ここで v_i は T_p の固有ベクトルであり、かつ μ は

$$\mu = \frac{2}{2 - \sigma_p} \quad (62)$$

で定義される。また z_i は

$$R^{(n)}(z) = T_n(z) / T_n(1/\mu)$$

で定義された $R^{(n)}$ の一次微係数の零点で

$$\frac{1}{\mu} > 1 \geq z_1 > z_2 \cdots > z_{n+1} \geq -1$$

の関係を満足するものである。そうすると、最終的には

$$\alpha^{(1)} = \mu,$$

$$\alpha^{(n)} = \frac{4}{\sigma_p} \frac{T_{n-1}(1/\mu)}{T_n(1/\mu)}, \quad n > 1 \quad (63)$$

$$\beta^{(1)} = 0,$$

$$\beta^{(n)} = \frac{T_{n-2}(1/\mu)}{T_n(1/\mu)} \quad (64)$$

の関係が導かれる。

以上の説明から明らかなように、チェビシェフ加速法の効果は優越比 σ_p に左右される。原子炉の固有値問題では、臨界から離れるにつれ、基本モードの優越性がくずれ、 $\sigma_p \rightarrow 1$ となるので、加速性が悪くなり、チェビシェフ加速が要求されてくる。

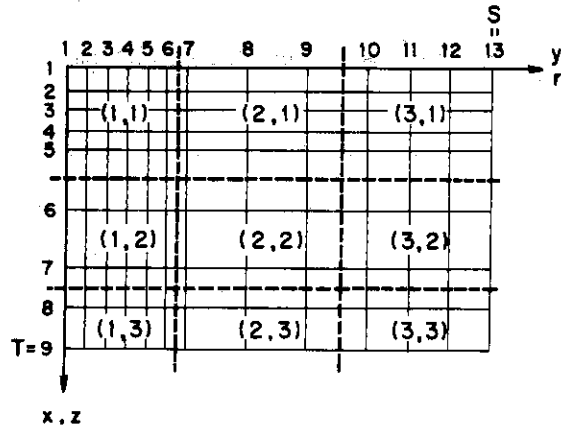


Fig. 1 Fine meshes and coarse mesh blocks (\bar{S}, \bar{T}) in GAMBLE-5

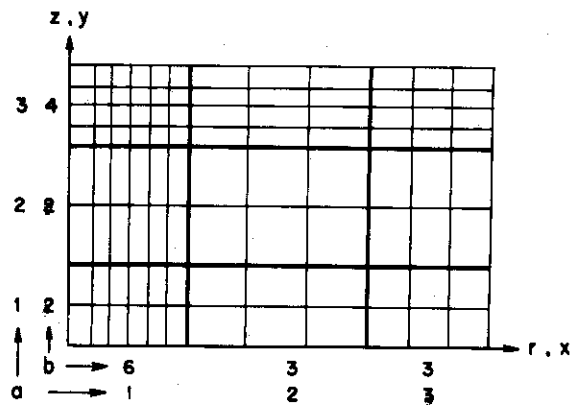


Fig. 2 Fine and coarse meshes in TWOTRAN

(a stands for the coarse mesh number, and b refers to the number of fine mesh intervals in each coarse mesh interval.)

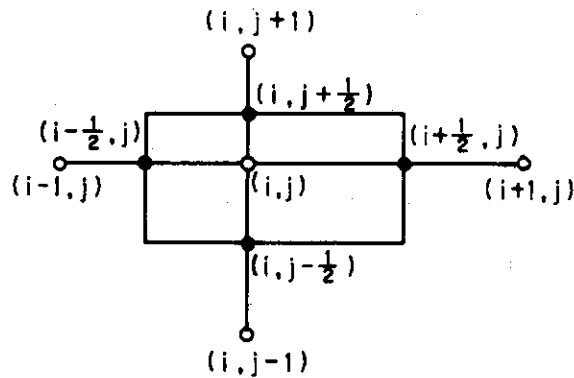


Fig. 3 Space mesh indexing scheme and unit cell in TWOTRAN

3. 数値実験による加速法の評価

行列方程式の反復解法の収束加速アルゴリズムの効果、特に多群多領域の原子炉核計算で用いた場合の各種アルゴリズムの性能の系統的な評価は行われていないが、散発的な数値実験による一部の加速アルゴリズムの総体評価の結果はいくつかの論文でみる事ができる。ここでは第2章で議論した加速アルゴリズムについての数値実験例をサーベイして、第2章の定性的な議論に定量的な議論を加味する。

3.1 再釣合法

先ず、粗メッシュ再釣合法から議論を始めよう。

2次元拡散計算におけるこの方法の評価については、Froehlich がGAMBLE-5を用いて行ったのが唯一と発表されている⁽⁸⁾。計算の対象とした体系及び計算に用いた定数の詳細は不明であるが、2次元(多分、直方体であろう)均質炉心で、エネルギーは3群、詳細メッシュの切り方は非常に不均一である。この場合は解析解が得られているので⁽³⁷⁾、反復解の誤差評価はこの解析解を基準として行っている。その結果はTable. 1にまとめて引用してあるが、粗メッシュ再釣合法を用いない時は、収束が遅く、中性子束の収束率が 10^{-1} 以下になるまでの反復回数は221回であり、その時の固有値 k_{eff} の誤差は依然として23%もある。一方、粗メッシュ再釣合法を用いると、同じ中性子束収束条件での反復回数は42回に過ぎず、その上 k_{eff} の誤差はわずか0.03%である。また計算時間も1/4になっている。計算時間が短縮されるのは、加速にもよるが、粗メッシュ上の反復計算は全ての計算機のコア・ストレージのみで行うことができ、補助記憶装置とデータのやりとりをする必要がないこともその原因となっている。

Table 1の一例のみでは一般的な結論は引き出し得ないし、加速の効果は詳細メッシュと粗メッシュの選び方にも依存するが、この方法は加速法として極めて有望であるということはできよう。また拡散方程式に対しては、この粗メッシュ再釣合法の数学的基礎は確立しているとみなせるので^{(3), (8)}、安心して使えるのも心強い。しかし、SN法にこの加速法を適用した場合には、まだ議論の余地が残っている。

2次元SNコードについては、Nakamura が全体系再釣合法、粗メッシュ再釣合法及びNakamura が定式化した新しいSN粗メッシュ加速法の比較を2領域xy体系について1群近似で行っている⁽⁵⁾。このNakamura の新しい方法については第4章で議論することとし、ここでは立入らないことにする。

1次元SN計算における再釣合法についてはReed⁽³⁸⁾が議論しているが、これについては第4章で取り上げることにする。

ここで、TWOTRAN型のSNアルゴリズムにおける粗メッシュ再釣合法についてのNakamura の計算結果を見てみよう。体系と核定数はFig. 4に示してある。詳細メッシュとしては同一巾で 24×24 の分割をし、粗メッシュとしてはやはり同一巾で、 6×6 と 12×12 の二通りの取り方をし、粗メッシュの数と収束性の関係をチェックしている。Fig. 5

は Fig. 4 の A 点における S_0 計算での中性子束の誤差と反復回数との関係を示したものであり、Fig. 6 は Z 点における誤差を示したものである。反復の初期の段階では全体系再釣合い法を用いた場合より粗メッシュ再釣合い法を用いた方が収束が早いですが、誤差が 0.1% 位になると、粗メッシュ再釣合い法の場合誤差の振動が生じ、収束が不安定になっている。拡散近似の場合には一様に収束するのに反して、 S_N アルゴリズムでは何故振動が生じるのかその原因は未だ解明されていない。全体系再釣合い法は反復の初期においては収束が遅いが、そのような振動は生じず、大体一様に収束していることが Fig. 5 と Fig. 6 から分る。TWOTRAN では、この問題を意識してかどうかわからないが、反復の初期段階では粗メッシュ再釣合い法を用いるが、これによる加速効果が失われてきたら、全体系再釣合い法を用いるようにプログラムされている。また Fig. 5 から分ることは、粗メッシュ数を増やすことは必ずしも精度の向上に単純には結びつかないということである。

2次元 S_N アルゴリズムでの粗メッシュ再釣合い法の振動現象を解明するには、誤差解析を行う必要があるが、未だその手法は確立されておらず、具体的に行われた例もない。ダイヤモンド差分法による負の中性子束の問題、粗メッシュ化した場合の誤差ベクトルの高次モードの減衰の乱れ等を明らかにすることから、この方法の改良の手掛りが得られるであろう。

2次元 P₃ 近似における粗メッシュ再釣合い法については、同じく Nakamura が Fig. 4 の体系について行っているので⁽⁵⁾、ここで紹介しておく。詳細メッシュは 24×24 、粗メッシュは 5×5 で、詳細メッシュ上での反復計算の 10 回目毎に粗メッシュ再釣合いを行っている。詳細メッシュ上での計算の加速には優緩和法を用いており、その加速因子の値としては 1.92 を最適値として用いており、粗メッシュ上の計算ではいずれに対しても 1.5 を用いている。内側反復の回数及び計算時間を Table 2 引用した。基本モードに対しては、粗メッシュ再釣合いを用いると、反復回数は 15.7%、計算時間は 17.1% になっている。

第 2 章で議論したその他の再釣合い法についての評価については公表されているものは見当たらない。

3.2 緩和法

緩和法による加速性は加速因子の値に強く依存することは前にも述べたが、(23) 式で定義された緩和法について Hageman が行った計算例⁽²³⁾を紹介しよう。計算の詳細は不明であるが、その結果の一部を Table 3 にまとめた。

Table 3 で β_b , $\tilde{\beta}_1$, $\tilde{\beta}_2$ は各々 ρ^2 , $\tilde{\rho}_1^2$, $\tilde{\rho}_2^2$ に対応している。スペクトル半径 ρ が 1 に近づくとつれ、その値の事前評価に要求される予測精度は厳しくなるが、最適値に近い値を用いた場合の加速性は極めて顕著である。

Hageman は、また、先に名前を挙げただけで議論は第 4 章に譲った Carre' の方法、Reid の方法、 $\sigma_1 - RQ$ 法についても数値的に相互評価をしている⁽²³⁾。これについては第 4 章で紹介する。他には Wachspress のテキストに簡単な報告があるが⁽³⁹⁾、ここには引用しない。

3.3 線緩和法

2次元拡散計算における線緩和法及びADI法についてはWachspressのテキストに簡単な計算例が報告されている⁽⁴⁰⁾。他に系統的な評価は報告されていないので、Wachspressの得た結論を簡単に紹介しておく。

Wachspressは2次元3群2領域問題で緩和法とADI法の比較をしているが、加速因子の値により緩和法の方が良い場合もあり、悪い場合もあり、一般的な結論は出し難いが、加速因子の最適値が比較的小さい問題に対してはADI法は良くないようである。

3.4 ウィーラント法

ウィーラント法についてはWachspress⁽⁴⁰⁾、Feldmark⁽²⁰⁾の報告があるが、単純なウィーラント法については、第3章で議論したように、あまり芳しい結果は得られていない。2次元均質体系についてのWachspressの計算結果を紹介しておこう。メッシュは 31×31 で、メッシュ巾は一様に $\Delta x = \Delta y = 1 \text{ cm}$ である。3群定数の値⁽⁴¹⁾はここには引用しないが、結果をTable 4にまとめておく。Table 4から分るように、ウィーラント加速をしない場合($\lambda_e = \infty$, $\lambda / \lambda_e = 0$)に比較して、固有値 λ とその見積り値 λ_e の比 λ / λ_e が0.86の場合反復回数が約半分になっている。一般にこの比が1に近づくにつれ、収束性は良くなるが、さらに1に非常に近づくと、収束性に乱れが生ずることもTable 4は示している。この乱れの発生の原因についての明確な議論はまだなされていない。このように λ_e の見積りの難しさがウィーラント法の欠点となっている。

Feldmarkの数値計算例についてはFeldmarkの改良型ウィーラント法(Modified Wielandt method)と一緒に第4章で議論する。GAMBLE-5で用いているアルゴリズムについての評価の報告はされていない。

3.5 チェビシエフ加速法

1次元SNアルゴリズムにおけるチェビシエフ加速については、Reedが未臨界炉について評価を行っている⁽³⁸⁾。計算に用いた定数はTable 5にまとめてあるが、問題M1とM2を比較すると、核断面積の加速性への影響が、M1とM3の比較から炉心サイズ、M3とM4からメッシュ巾の影響が分る。M5とM6は吸収が大きく、したがって未臨界性の大きい場合である。チェビシエフ加速を行った場合と行わない場合の反復回数をTable 6にまとめた。M1~M4ではチェビシエフ加速を用いると、反復回数はメッシュ巾、炉心サイズ、断面積にほとんど関係なく、17~18%に減少する。先にも触れたように、未臨界性の大きい炉の場合(M5~M6)は、加速性は悪く、あまり効力はない。

1次元拡散計算におけるチェビシエフ加速についてはFroehlichの評価例がある⁽⁸⁾。これは1群近似でチェビシエフ加速をした時としない時の外側反復回数を比較したもので、計算の対象として熱出力300MWの各種動力炉、すなわち、高温ガス炉(HTGR)、沸騰水型軽水炉(BWR)、加圧水型軽水炉(PWR)、ガス冷却高速炉(GCFR)、流体金属冷却高速増殖炉(LMFBR)を選んでいる。Table 7にその結果をまとめてあるが、軽水炉の場合は特に加速効果が顕著で、チェビシエフ加速を用いると、外側反復回数は1.3~1.4%にな

る。高温ガス炉ではやや効果が軽水炉の場合より落ちるが、それでも4%になる。高速炉の場合は熱中性子炉より加速性が落ち、GCFRで7.2%、LMFBRで6.8%である。これは高速炉における優越比 σ_p は熱中性子炉の場合より1に近いためである。また、軽水炉の場合、外側反復の回数が非常に多いのは、拡散距離(diffusion length)単位で測った炉心の大きさは、他の型の炉に比較して極端に大きく、それだけ多くのメッシュ点をとる必要があるためである。なお、Froehlichが計算に用いた具体的なデータの値は不明である。

2次元計算では理由ははっきりしないが、チェビシェフ加速の効き方が悪い。その実例としてNakamuraがBWR炉心に対してxy-2次元拡散計算におけるチェビシェフ加速と粗メッシュ再釣合いの効果を比較している⁽⁴⁾のを示そう。計算に用いた炉心の構造及び3群定数はFig. 7にまとめた。詳細メッシュは50×50、粗メッシュは8×8である。

Nakamuraのチェビシェフ加速アルゴリズムは、2.7節のものと若干型式が異なるので、ここに簡単にまとめておく。(55)式を次の反復アルゴリズムで解く。

$$\phi^{(n)} = A^{-1} S^{*(n-1)},$$

$$S^{*(n)} = \theta^{(n)} S^{(n)} + (1 - \theta^{(n)}) S^{*(n-1)} \quad (65)$$

$$S^{(n)} = \frac{1}{\lambda^{(n)}} B \phi^{(n)}$$

$$\lambda^{(n)} = (\phi_{\omega} S \phi^{(n)}) / (\phi_{\omega} S^{*(n-1)}).$$

ここで、 θ はチェビシェフ多項式で与えられるパラメーター、 ϕ_{ω} は重み関数である。源の誤差を次のように定義する。

$$E_s^{(n)} = \max_{ij} (S_{ij}^{(n)} / S_{ij}^{*(n-1)}) - \min_{ij} (S_{ij}^{(n)} / S_{ij}^{*(n-1)}). \quad (66)$$

ここで ij は詳細メッシュ・インデックスである。この源誤差の減衰を反復回数を横軸にして示したのが、Fig. 8で、6次までのチェビシェフ多項式を用いた場合と粗メッシュ再釣合いを行った場合の比較をしてある。チェビシェフ加速法を用いた場合の収束の悪さがこの図から明らかである。

Table 1. Effectiveness of coarse mesh rebalancing method in 3-group homogeneous reactor problem, with highly non-uniform mesh, tested by GAMBLE-5⁽⁸⁾.

	Iterations	Flux convergence	k_{eff} error	Execution time (UNIVAC 1108)
No rebalance	221	0.0001	23 %	4 min
With rebalance	42	0.0001	0.03 %	1 min

Table 2. Comparison of inner iteration numbers and execution times with and without coarse mesh rebalancing in P_3 algorithm⁽⁵⁾.

Flux	Acceleration factor β of overrelaxation	Without		With	
		No. of iterations	Execution time (sec)	No. of iterations	Execution time (sec)
ϕ_1	1.96	102	4.69	16	0.80
ϕ_2	1.92	79	3.64	30	1.55

IBM370/165.

Flux convergence criteria: 10^{-4} .

Table 3. Sensitivity of the SOR method to the acceleration parameter⁽²³⁾ for the error reduction by a factor of 10

(Spectral radius) ² ρ^2	Estimate for ρ^2		Acceleration parameter			No. of iterations			
	$\tilde{\rho}_1^2$	$\tilde{\rho}_2^2$	$\tilde{\rho}_b$	$\tilde{\rho}_1$	$\tilde{\rho}_2$	$I(\tilde{\rho}_b)$	$I(\tilde{\rho}_1)$	$I(\tilde{\rho}_2)$	no acceleration
0.98	0.9799	0.97	1.7522	1.7517	1.7047	8	9	16	114
0.999	0.9989	0.989	1.9387	1.9358	1.8101	34	50	236	4,602
0.9999	0.9998	0.9899	1.9802	1.9721	1.8174	115	278	2,308	46,050
0.99999	0.99989	0.98999	1.9937	1.9792	1.8181	364	2,360	23,032	460,515

Table 4. Wielandt iterations for a two-dimensional homogeneous reactor⁽²⁾

ADI sweeps per outer iteration	Inner iteration error	λ/λ_e	No. of outer iterations for error in $\lambda < 5 \times 10^{-4}$
20	0.01	0	7
40	0.01	0.86	4
50	0.01	0.97	Error $\approx 10^{-2}$ after 4 iterations

Table 5. Model problem characteristics for the evaluation of effectiveness of the Chebyshev acceleration in one-dimensional S_2 algorithm⁽³⁹⁾

Problem	Secondary neutrons per collision	Macroscopic c.s. Σ (cm ⁻¹)	Mesh size Δx (xcm)	Core thickness R (cm)
M1	1	1.0	1.0	30.0
M2	1	2.0	1.0	30.0
M3	1	1.0	1.0	60.0
M4	1	1.0	2.0	60.0
M5	2/3	3.0	1.0	30.0
M6	1/2	4.0	1.0	30.0

Table 6. Effectiveness of the Chebyshev acceleration in one-dimensional S_2 algorithm⁽³⁹⁾

Problem	No. of iterations to convergence	
	Unaccelerated	Chebyshev acceleration
M1	1058	197
M2	2628	454
M3	2631	455
M4	2628	454
M5	22	13
M6	14	10

Convergence criteria : 10^{-4} .

Table 7. Effectiveness of the Chebyshev acceleration in one-dimensional diffusion algorithm⁽⁸⁾. Necessary number of outer iterations for error reduction by a factor of 10^3

Reactor type 300 MW(t)	Straight outer iterations	Chebyshev acceleration
HTGR	940	38 (4 %)
BWR	7550	105 (1.4 %)
PWR	8650	112 (1.3 %)
GCFR	278	20 (7.2 %)
LMFBR	311	21 (6.8 %)

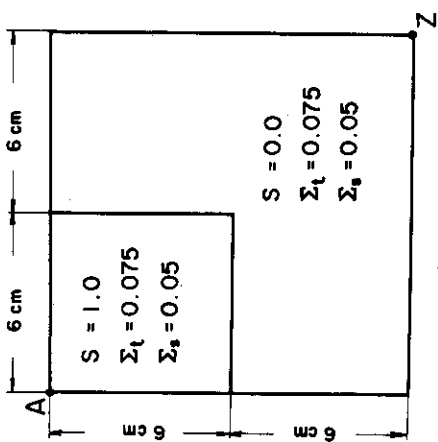


Fig. 4 Characteristics of two dimensional problem (5)

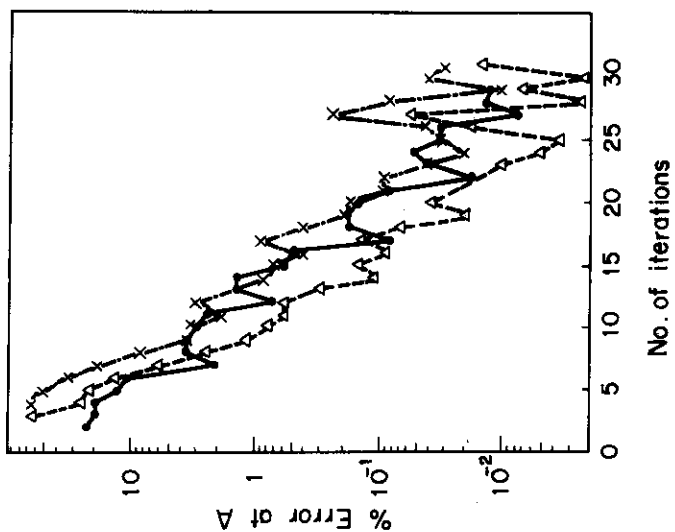


Fig. 5 Comparison of decay of flux errors at A in Fig.4 for whole system and coarse mesh rebalancing methods (5)

- whole system rebalancing
- △ coarse mesh rebalancing (6 x 6)
- x coarse mesh rebalancing (12 x 12)

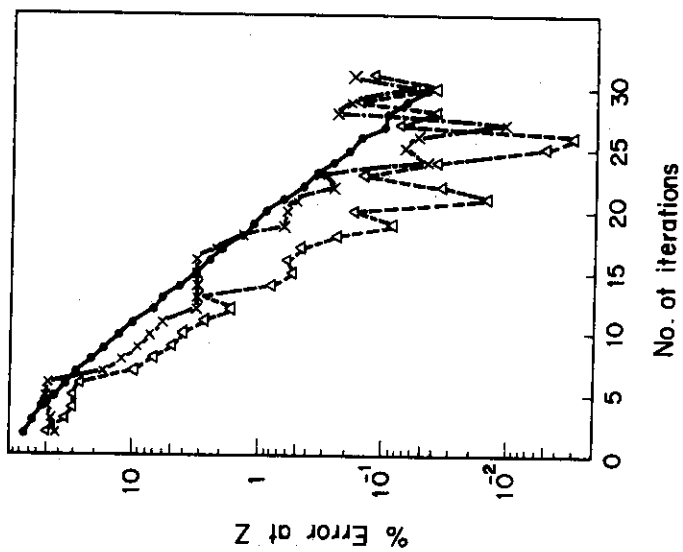


Fig. 6 Comparison of decay of flux errors at Z in Fig.4 for whole system and coarse mesh rebalancing methods (5)

- whole system rebalancing
- △ coarse mesh rebalancing (6 x 6)
- x coarse mesh rebalancing (12 x 12)

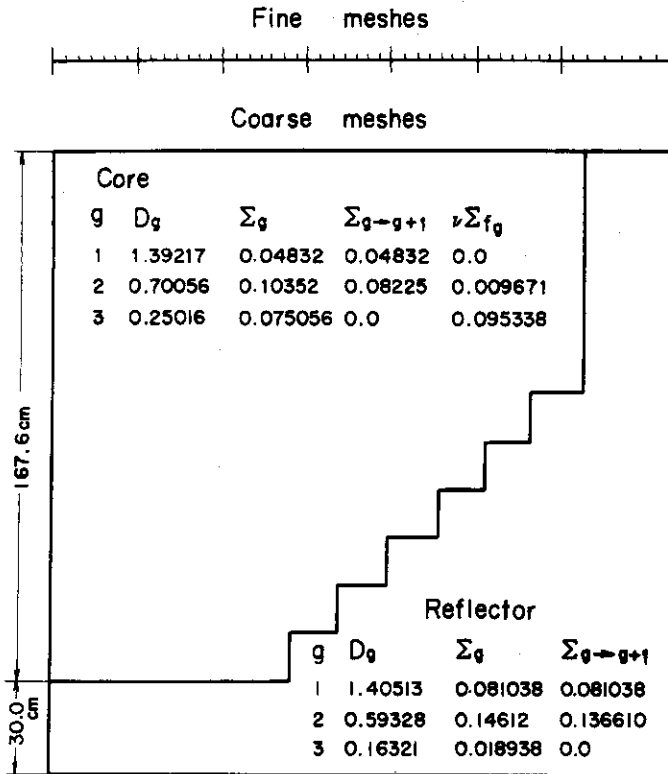


Fig. 7 Model problem characteristics for the evaluation of the Chebyshev acceleration method in two-dimensional diffusion algorithm⁽⁴⁾

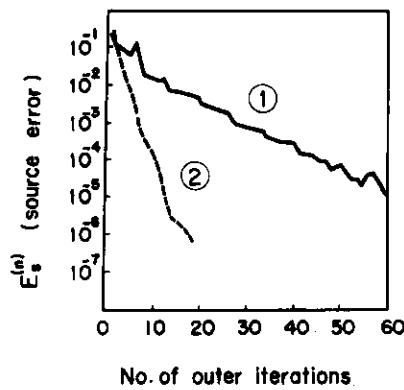


Fig. 8 Comparison of decay of source errors in two dimensional diffusion algorithm⁽⁴⁾

- 1) 6-th order Chebyshev
- 2) coarse mesh rebalancing

4. 核計算コードにおける加速法の改良

前章までにおいては既存の核計算コードにおいて既に採用されている種々の加速法について議論してきたが、この章では加速法の最近の研究をサーベイし、それも含めて我々の手持ちの核計算コードの収束加速の改良の方法について議論する。

4.1 1次元 S_N コードの改良

今までの議論から明らかなように、1次元 S_N アルゴリズムにおいては粗メッシュ再釣合法とチェビシェフ加速法は非常に有効であり、1DFX及びDTF69ではチェビシェフ加速を行えるようになってきていることも述べた。1次元 S_N コードで粗メッシュ再釣合法を用いているものは未だ公開されていないが、1次元 S_N アルゴリズムにおけるこの加速法の効果についてはNakamura⁽⁴⁾とReed⁽³⁸⁾の計算例がある。またエネルギー群についての再釣合法についてはClancy & Donnellyの試みがある⁽⁴²⁾。これら粗メッシュ再釣合法、チェビシェフ加速法、エネルギー群再釣合法は一般に流布しているDTF-IVやANISNでは用いられていないので、先ず考えられることは、これらの方法を使えるようにコードを改造することである。

次にReedの再釣合法に関する改良とその応用例を紹介しよう。

1次元平板系に対する S_N アルゴリズムをダイヤモンド差分法と組合せて次の式で定義する(38)。

$$\mu_m \frac{N_{m,i+1/2}^{(k+1)} - N_{m,i-1/2}^{(k+1)}}{\Delta x_i} + \Sigma_{m,i}^{(k+1)} = c \Sigma \phi_i^{(k)} + q_i,$$

$$N_{m,i}^{(k+1)} = \frac{N_{m,i+1/2}^{(k+1)} + N_{m,i-1/2}^{(k+1)}}{2} \quad (67)$$

ここで k は外側反復の番号で、 q_i は外部源、 ϕ_i は

$$\phi_i = \sum_m w_m N_{m,i} \quad (68)$$

で与えられる中性子束である。その他の記号の意味は既に用いられているものと同じである。再釣合のための補正因子 f_i を用いた中性子の釣合いを次のように、角度について積分して、設定する。すなわち、

$$\langle M_m \rangle_{\pm} \equiv \sum_{\substack{\mu_m > 0 \\ <}} w_m M_m, \quad (69)$$

の定義を用いて、

$$\begin{aligned}
 & - \left[\langle \mu_m N_{m, i+1/2}^{(k)} \rangle^- + \kappa_{i+1/2} \right] f_{i+1} \\
 & - \left[\langle \mu_m N_{m, i-1/2}^{(k)} \rangle^+ + \kappa_{i-1/2} \right] f_{i-1} \\
 & + \left[\langle \mu_m N_{m, i+1/2}^{(k)} \rangle^+ + \langle \mu_m N_{m, i-1/2}^{(k)} \rangle^- + \kappa_{i+1/2} + \kappa_{i-1/2} \right. \\
 & \left. + (1-e) \Sigma \Delta x \phi_i^{(k)} \right] f_i = \bar{q}_i \Delta x. \tag{70}
 \end{aligned}$$

ここで $\kappa_{i+1/2}$ は Reed が導入した任意パラメターで、Reed は次のように選んでいる。

$$\kappa_{i+1/2} = X \left[\frac{e \Sigma_{i+1/2} \Delta x}{4} \max(\phi_i^{(k)}, \phi_{i+1}^{(k)}) \right], \tag{71}$$

ここで X は任意定数である。また \bar{q}_i は

$$\bar{q}_i = \sum_m w_m q_m$$

で与えられる。(70) 式では粗メッシュ化は行われておらず、各メッシュ点毎の f_i を求める形になっている。(70) 式で全ての $\kappa_{i+1/2}$ を 0 にすると、通常の再釣合い法と同一になる。

Reed は Table 5 の問題 M1 ~ M5 について、任意パラメター X の値を 0 と 4 の間で変化させて、 S_0 計算を行い、粗メッシュと詳細メッシュの数が等しい場合の収束に要した反復回数を比較し、 $X=1$ の時が一番収束が良いという結論を得た。Table 8 から分るように、 κ というパラメターを (70) 式に導入した効果は特に問題 M2 と M4 に見られる。この場合通常の再釣合い法 ($X=0$) では収束しないが、 $X \neq 0$ を用いると、この場合にも収束している。粗メッシュ化を行った場合についての評価はされていない。

次にエネルギー群についての再釣合い法を考えてみよう。エネルギー群毎の中性子束のスクーリングは ANISN では行いようになっているが、補正された中性子束は釣合いの条件を満足しない。Clancy & Donnelly は中性子の釣合いが保存される形にしたエネルギー依存再釣合い因子を定義し、これによる加速性の評価を行っている⁽⁴²⁾。Clancy & Donnelly の場合は、空間的には一定のエネルギー群毎の補正因子 C_g を用いた釣合いの式を次のようにとる。

$$\int dV \left\{ - \sum_{g' < g} \Sigma_{g' \rightarrow g}^{(n)} \phi_{g'} + [Q_g + \sum_{g' < g} \Sigma_{g' \rightarrow g}^{(n)} \phi_{g'} + \sum_{g > g} \Sigma_{g \rightarrow g'}^{(n-1)} \phi_{g'} \right.$$

$$+ \alpha \sum_{g \rightarrow g'} \left(\phi_g^{(n)-} - \phi_{g'}^{(n)} \right) \} C_g - \sum_{g' > g} \sum_{g' \rightarrow g} \phi_{g'}^{(n)} C_{g'} \} = \int d v Q_g \quad (72)$$

ここで

$\int Q_g d v = g$ 群への中性子利得 (核分裂中性子も含む)

である。定数 α は g 群内の中性子釣合いを保証するためのもので、 $\phi_g^{(n)-}$ は n 回目の外側反復時の最後の内側反復計算をする前の値で、内側反復補正をする場合は $\alpha = 0$ として、群内散乱を除去し、内側反復補正をしない時は $\alpha = 1$ とする。

((72) 式を解くのに Clancy & Donnelly は Gauss-Jordan の消去法⁽⁴³⁾ を用いているが、この方法が最適かどうかは検討の余地がある。1次元平板6群 S_2 計算の結果によると、上方散乱の寄与が大きい場合ほど加速効果があるが、 C_g の正值性の問題 (C_g に負の解がでたら良くない) はさらに種々の体系でテストしてみる必要があり、また、遮蔽計算のような問題に適用するには、 C_g に空間依存性を持たせることも重要であろう。Clancy & Donnelly の計算は分かり易い形にまとめられていないので、ここには引用しない。

4.2 2次元 S_N コードの改良

2次元 S_N コードで改良すべき点として直ぐに気が付くことは、TWO TRAN の場合では、エネルギー群についての粗メッシュ再釣合い法を適用することである。ただし、第3章で示したように空間メッシュについての粗メッシュ再釣合い法では、収束がある段階まで進むと、誤差の振動が生じ、収束性に乱れがでるが、群メッシュについての粗メッシュ再釣合いをこれに加えた場合、つまり、空間・エネルギーについての GAMBLET-5 のような粗メッシュ再釣合いをを行った時、この誤差の振動がどうなるか、低減されるか、増幅されるか、興味のある問題である。もし、振動が生じるならば、空間の取扱いの時と同様に、収束がある程度進んだところで、全エネルギー群一因子再釣合いに切り換えればよい。

角度空間にも粗空間を持ち込む試みは Nakamura が行っている⁽⁵⁾。 S_N 計算で求めた未収束の中性子角度分布 (angular flux) を $N_{i,j}^m$ としよう。ここで $m, (i, j)$ は各々、角度空間、 $x-y$ 座標空間における詳細メッシュのインデックスである。そこで、次の再釣合い補正

$$N_{i,j}^m f(x_i, y_j, \Omega)$$

を行うとする。 Ω 空間を四つの粗空間に分割し、再釣合い関数を各粗空間毎に

$$f(x, y, \Omega) = f_{k,l}^a, \quad a = 1, 2, 3, 4 \quad (73)$$

とする。ここで、 (k, l) は座標空間の粗メッシュブロックであり、

$$x_k < x < x_{k+1}, \quad y_l < y < y_{l+1}$$

とする。また、 a を Ω を射影した $x-y$ 面の象限 (quadrant) とすると、(73) 式で

$$\Omega \in a$$

である。第一象限における釣合いの式は次のように書ける。

$$\begin{aligned} & \left[J_{x,k,l}^1 + J_{y,k,l}^1 + A_{k,l}^1 \right] f_{k,l}^1 = \sum_{\beta=1}^4 B_{k,l}^{\beta \rightarrow 1} f_{k,l}^\beta \\ & + J_{x,k-1,l}^1 f_{k-1,l}^1 + J_{y,k,l-1}^1 f_{k,l-1}^1 + S_{k,l}^1, \end{aligned} \quad (74)$$

ここで、 D_{kl} で粗メッシュブロックを表わし、 I, J は各々 D_{kl} の右境界及び上境界のインデックスとすると、

$$J_{x,k,l}^\alpha = \sum_{J \in D_{kl}} \Delta y_J \sum_{m \in \alpha} \Omega_x^m w_m V_{i,j}^m \quad (75)$$

$$J_{y,k,l}^\alpha = \sum_{I \in D_{kl}} \Delta x_I \sum_{m \in \alpha} \Omega_y^m w_m H_{i,j}^m$$

と書ける。ここで、 V と H は離散化された中性子角度分布の x, y 成分、 w_m は角度の重みである。(74) 式の $A_{k,l}^1$ は再釣合いを行う前の第一象限内に方向を持つ中性子の全吸収率、 $B_{k,l}^1$ は散乱率であり、ダイヤモンド差分法の代りに、平均中性子束を

$$N_{i,j}^m = \frac{1}{4} (H_{i,j}^m + V_{i,j}^m + H_{i+1,j}^m + V_{i,j+1}^m) \quad (76)$$

で定義すると、一般に α 象限に対しては、

$$A_{k,l}^\alpha = \sum_{i,j \in D_{kl}} \Delta x_i \Delta y_j \sum_{m \in \alpha} \Sigma_{t,i,j} w_m N_{i,j}^m \quad (77)$$

$$B_{k,l}^{\beta \rightarrow \alpha} = \sum_{i,j \in D_{kl}} \Delta x_i \Delta y_j \sum_{n \in \alpha} \sum_{m \in \beta} \Sigma_{s,i,j}^{m \rightarrow n} w_m N_{i,j}^m \quad (78)$$

である。同じく、源は

$$S_{k,l}^\alpha = \sum_{i,j \in D_{kl}} \Delta x_i \Delta y_j \sum_{m \in \alpha} w_m q_{i,j}^m \quad (79)$$

となる。この Nakamura のアルゴリズムで

$$\phi_{k,l}^1 = \phi_{k,l}^2 = \phi_{k,l}^3 = \phi_{k,l}^4$$

とおくと、TWO TRAN で用いられている粗メッシュ再釣合い法と同じになる。

Nakamura は Fig. 4 の問題でこの方法の S_0 によるテストを行い、角度について粗メッシュ化を行わない場合と比較している。空間メッシュについては同じ粗メッシュ 6×6 をとった場合を Fig. 9 と Fig. 10 に示した。粗メッシュ再釣合いは詳細メッシュ上の反復計算 5

回目毎に行っている。角度の粗メッシュ化を用いると、誤差0.1%に低減するまでの反復回数は、これを用いない場合の半分位になるが、誤差の振動は増幅される傾向にあり、収束がある段階に到ると、加速効果が失われることがFig. 9とFig. 10から分る。

4.3 2次元拡散コードの改良

2次元拡散コードにおいては5点差分アルゴリズムの枠内では改良の余地はあまり残されていないように思われる。しかし、2次元拡散コードで収束加速に用いられている緩和法やウィーラント法の効果はその加速因子や固有値の見積り値に大きく左右されるので、これらの最適値をアプリアリに求める方法の研究は盛んに行われており、これらの研究の成果を拡散計算にとり入れることは極めて重要である。したがって、この節では上記の観点によるまとめを行う。

(1) 緩和法における加速因子の評価

この問題に関してはHagemanの報告⁽²³⁾がよくまとまっている。(19)式の行列方程式

$$AX = k \quad (19)$$

を考えよう。ここで、Aは部分行列 A_{ij} により、次のように表示できるとする。

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1N} \\ A_{21} & A_{22} & \cdots & A_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ A_{N1} & A_{N2} & \cdots & A_{NN} \end{bmatrix} \quad (80)$$

ここで重要な概念を二つ定義しておこう⁽⁴⁴⁾。

定義：行列Aが(80)のように分割できたとした時、 $A_{ij} \neq 0$ ならば、 $r_i - r_j = 1$ または -1 なる成分を持つ配列ベクトル(ordering vector) $r = (r_1, \dots, r_N)$ が存在する時、Aは2・サイクル行列という。

定義：行列Aが2・サイクル行列の時、 $A_{ij} \neq 0$ ならば、 $r_j - r_i = 1$ ($j > i$)及び $r_j - r_i = -1$ ($i > j$)なる配列ベクトルrが存在するならば、Aは首尾一貫して配列されている(consistently ordered)という。

一方、Aを(20)式のように分解した時

$$J \equiv D^{-1} (L + U)$$

を(22)式のヤコービ反復行列と呼ぶ。(23)式を次のように書く。

$$X^{(q+1)} = L_{\beta} X^{(q)} + \beta (D - \beta L)^{-1} k \quad (81)$$

ここで

$$L_{\beta} \equiv (D - \beta L)^{-1} [\beta U + (1 - \beta) D] \quad (82)$$

はSOR反復行列と呼ばれる。

行列Aが2サイクルでかつ首尾一貫して配列されるならば、 L_β の固有値 λ とJの固有値 μ は

$$(\lambda + \beta - 1)^2 = \lambda \beta^2 \mu^2 \quad (83)$$

の関係にあり、さらに μ が実数で、Jのスペクトル半径 $\rho(J)$ が1より小さい(<1)ならば、

$$\beta_b = \frac{2}{1 + \sqrt{1 - \rho^2(J)}} \quad (84)$$

とすると

$$\rho(L_\beta) = (\beta - 1), \quad \beta \geq \beta_b \text{ の時} \quad (85)$$

$$\rho(L_{\beta_b}) < \rho(L_\beta), \quad \beta = \beta_b \text{ の時}$$

が成り立ち、 $\beta = \beta_b$ は収束性を最適化する(45)。

スペクトル半径 $\rho(J)$ を評価するいくつかの方法と数値実験例を紹介しておこう。これらの方法はいずれも、 β の過大評価値から値を低くして行って β_b に近づけるEXTERMINATORやCITATIONの方法とは逆に、 β の過小評価値から出発して、値を段階的に大きくして β_b に近づけるアルゴリズムである(46)。

Carre'の方法

残差ベクトル $R^{(q+1)} = X^{(q+1)} - X^{(q)}$ を用いると、(81)式は

$$R^{(q+1)} = L_\beta R^{(q)}$$

と書ける。 $\beta < \beta_b$ ならば、 L_β のスペクトル半径は

$$\lim_{q \rightarrow \infty} \frac{\|R^{(q)}\|}{\|R^{(q-1)}\|} = \rho(L_\beta) \quad (86)$$

で与えられる。ここで $\| \cdot \|$ はノルム記号である。したがって、 $\beta < \beta_b$ のとき、

$$Q(q) = \frac{\|R^{(q)}\|}{\|R^{(q-1)}\|} \quad (87)$$

は $\rho(L_\beta)$ の評価値として使える。(83)式から

$$\rho_q^2(J) = \frac{[Q(q) + \beta - 1]^2}{\beta^2 Q(q)} \quad (88)$$

によって $\rho(J)$ が求まり、 β の評価値

$$\beta(q) = \frac{2}{1 + \sqrt{1 - \rho_q^2(J)}} \quad (89)$$

が得られる。Carre のアルゴリズムは次の手順で計算を行う。

- (a) $\beta = 1.0$ として最初の 1 回の反復計算をする。
- (b) 次に $\beta = 1.375$ として 12 回反復計算をし、そこで、(87)、(88)、(89) 式により新しい β の値 β_e を決める。
- (c) $\beta = \beta_e - (2 - \beta_e) / 4$ として、また 12 回反復計算をし、再び (88) ~ (89) 式により β_e を決め、前回の値との差を $\Delta \beta_e$ とする。
 $\Delta \beta_e / (2 - \beta_e) < 0.05$ の時は、次の (d) へ進み、 > 0.05 の時は、(c) の計算を繰返す。
- (d) $\beta = \beta_e$ として SOR 計算を行う。

Reid の方法

$$E(s) = \begin{bmatrix} s^{\gamma_1} I & & & & \\ & s^{\gamma_2} I & & & 0 \\ & & & & \\ & & & & \\ 0 & & & & s^{\gamma_N} I \end{bmatrix}$$

とする。ここで s は J の固有値に関係した数で

$$s^2 - \beta \mu s + (\beta - 1) = 0$$

を満足する。 γ_i は配列ベクトル、 I は単位行列である。

$$y(q) = E^{-1} Q(q)^{1/2} R(q) \quad (90)$$

として、

$$\rho_q(J) = \frac{[y(q)]^T (L + L^T) y(q)}{[y(q)]^T D y(q)} \quad (91)$$

で、スペクトル半径の評価を行う。

$$\rho_q(J) \leq \rho(J)$$

であるから、 ρ_q から求めた β に対しては

$$\beta(q) \leq \beta_b$$

が成り立つ。Reid の方法では A は正値 (positive definite) であることが必要である。具体的には次の手順で計算を行う。

- (a) $\beta = 2.0$ として、2回反復計算をする。
 (b) (87), (90), (91), (89)式により新しい β の値 β_c を求める。
 (c) $\beta = \beta_c$ として12回反復計算を行う。その後で $C = [Q(q)]^6 / (\beta_c - 1)^5$ を計算し、 $C > 1$ ならば、(b)、 $C < 1$ ならば(c)の計算を行う。

σ_1 - RQ法

まず、ここで必要な概念の定義をしておく。2-サイクル行列Aを

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

とした時、Aは σ_1 2サイクル型に分割されたという。

(19)式を σ_1 2サイクル型表示で

$$\begin{bmatrix} D_1 & B \\ B^T & D_2 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$

と書く。(92)式に対する緩和式は次のように書くことができる。

$$X_1^{(q+1)} = \beta \{ -D_1^{-1} B X_2^{(q)} + D_1^{-1} k_1 - X_1^{(q)} \} + X_1^{(q)}, \quad (93)$$

$$X_2^{(q+1)} = \beta \{ -D_2^{-1} B^T X_1^{(q+1)} + D_2^{-1} k_2 - X_2^{(q)} \} + X_2^{(q)}.$$

残差ベクトル

$$R_2^{(q+1)} = X_2^{(q+1)} - X_2^{(q)}$$

と定義すると、 $R_2^{(q+1)}$ に対応する Gauss-Seidel 行列 L_β の表現 L_2 は、(93)式から

$$L_2 = \begin{bmatrix} O & -D_1^{-1} B \\ O & -D_2^{-1} B^T D_1^{-1} B \end{bmatrix} \quad (94)$$

となることが分る。すると、

$$\rho^2(J) = \rho(L_2)$$

の関係の証明することができ、したがって

$$\rho_q(L_2) = \frac{[R_2^{(q)}]^T B^T D_1^{-1} B R_2^{(q)}}{[R_2^{(q)}]^T D_2 R_2^{(q)}} \quad (96)$$

をスペクトル半径の評価値として用いることができる。具体的な計算の手順はReidの方法の場合と全く同じにする。 σ_1 -RQ法の特徴は $\rho(L_\beta)$ を直接評価する必要のないことである。

CP法 (Chebyshev polynomial method)

CP法は次の緩和反復アルゴリズムで定義される(47)。

$$X^{(q)} = \sigma_q [JX^{(q-1)} + D^{-1}k - X^{(q-2)}] + X^{(q-2)} \quad (97)$$

ここで、加速因子 σ_q はチェビシェフ多項式 T_q により与えられるもので、

$$\sigma_1 = 1,$$

$$\sigma_q = \frac{2}{\rho} T_{q-1}\left(\frac{1}{\rho}\right) / T_q\left(\frac{1}{\rho}\right), \quad q > 1,$$

であり、 ρ はヤコービ反復行列 J のスペクトル半径である。 D は(21)式で定義したものである。

残差ベクトル $R^{(1)}$ を J の固有ベクトル y で展開し、

$$R^{(1)} = \sum_{i=1}^n C_i y_i \quad (98)$$

とする。 r を $\rho(J)$ の評価値とすると、

$$R^{(q)} = \sum_{i=1}^n C_i P_{q-1}(\mu_i, r) y_i, \quad q > 1 \quad (99)$$

とすることができる。ここで P はチェビシェフ多項式から

$$P_q(z, r) \equiv T_q(z/r) / T_q(1/r)$$

の関係で定義されるもので、 μ_i は J の固有値である。

$$Q(2q+1) = \frac{\|R^{(2q+1)}\|}{\|R^{(2q-1)}\|} \quad (100)$$

を定義すると、

$$\lim_{q \rightarrow \infty} Q(2q+1) = \rho^2(J) \quad (101)$$

が成り立つ。

次に ρ の決め方であるが、 ρ の評価値として r を用いたCP計算を $(k+1)$ 回目の反復時に始めたとし、 $R^{(k+1)}$ を(98)式のように展開し、 $R^{(k+q+1)}$ を(99)式のように

展開すると、 $\gamma < \rho$ の時は

$$\lim_{q \rightarrow \infty} P_q(\mu_i, \gamma) / P_q(\rho, \gamma) = 0$$

であるから、 q が十分大きい時には

$$Q(k+2q+1) \simeq \frac{P_{2q}(\rho, \gamma)}{P_{2q-2}(\rho, \gamma)} \quad (102)$$

が得られ、 ρ の評価値として

$$\rho \simeq \gamma \cosh \left[\frac{\cosh^{-1} \left\{ T_{2q} \left(\frac{1}{\gamma} \right) V(k+2q+1) \right\}}{2q} \right] \quad (103)$$

を用いることができる。ここで $T_q(\omega)$ の定義

$$T_q(\omega) = \cosh [q \cosh^{-1} \omega]$$

及び

$$V(k+2q+1) \equiv \frac{\|R(k+2q+1)\|}{\|R(k+1)\|}$$

を用いた。(102)式で $\rho = \gamma$ とした時を $TQ(k+2q+1)$ とすると、収束率の評価として

$$C(k+2q+1) = \frac{\ln Q(k+2q+1)}{\ln TQ(k+2q+1)} \quad (104)$$

を用いることができる。

γ の値については数値実験の結果次のような制限をおくと良いとされている。

$$\begin{aligned} \gamma_1 &\leq 0.9486, \quad \gamma_2 \leq 0.985, \quad \gamma_3 \leq 0.99, \\ \gamma_4 &\leq 0.995, \quad \gamma_5 \leq 0.9975, \quad \gamma_6 \leq 0.9995, \\ \gamma_s &\leq 0.99999 \quad (s \geq 7). \end{aligned} \quad (105)$$

CP法のアルゴリズムは次の通りである。

- ヤコービ反復 ($\sigma_p = 1$) を 3 回行い、(100)、(101)式から最初の $\rho(J)$ の評価値 γ_1 を求める。 $\gamma_1 > 0.9486$ だったら、 $\gamma_1 = 0.9486$ とする。
- 評価値 γ_1 を用いて、9回チェビシエフ反復を行う。
- (104)式により C を計算する。 $C > 0.8$ ならばさらに2回チェビシエフ反復をし、 C の値をチェックする。 $C < 0.8$ になったら、 γ_2 を(103)式から求める(103)式で γ_1 とした ρ の値)。もし、 γ_2 が(105)式の条件に合わなかったら、 $\gamma_2 = 0.985$ とする。(b)の計算にもどる。以後の γ_s によるチェビシエフ反復回数には次の制限を設ける。

$$q_2 \geq 9, \quad q_3 \geq 15, \quad q_4 \geq 19,$$

$$q_s \geq 19, \quad q_s \geq 29 \quad (s \geq 6).$$

CCP法 (Cyclic Chebyshev Polynomial Method)⁽⁴⁸⁾

この方法は2次元拡散動特性コードTWIGLE⁽⁴⁹⁾で内側反復の加速に用いられている。CP法の改良されたもので、 $\sigma_1 - 2$ -サイクル表現を(92)式より一般的な型にとれるとし、

$$\begin{bmatrix} D_1 & B_1 \\ B_2 & D_2 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \quad (106)$$

とおく。CCPの反復アルゴリズムは次の式で定義される。

$$\begin{aligned} X_1^{(q)} &= \sigma_{q,1} \{ V_1^{(q)} - X_1^{(q-1)} \} + X_1^{(q-1)}, \\ D_1 V_1^{(q)} &= -B_1 X_2^{(q-1)} + k_1, \\ X_2^{(q)} &= \sigma_{q,2} \{ V_2^{(q)} - X_2^{(q-1)} \} + X_2^{(q-1)} \\ D_2 V_2^{(q)} &= -B_2 X_1^{(q-1)} + k_2, \end{aligned} \quad (107)$$

ここで

$$\begin{aligned} \sigma_{1,1} &= 1, \\ \sigma_{1,2} &= 2 / (2 - \rho^2(J)), \\ \sigma_{q,1} &= \frac{1}{1 - \frac{\rho^2(J)}{4} \sigma_{q-1,2}}, \quad (q > 2) \\ \sigma_{q,2} &= \frac{1}{1 - \frac{\rho^2(J)}{4} \sigma_{q,1}} \end{aligned} \quad (108)$$

である。残差ベクトルを

$$R_2^{(q)} \equiv V_2^{(q)} - X_2^{(q-1)}$$

で定義し、

$$Q(q) \equiv \frac{\|R_2^{(q)}\|}{\|R_2^{(q-1)}\|} \quad (109)$$

とする。(104)式に対応する収束率の評価は

$$C(k+q+1) = \frac{\ln \left[\frac{\delta_q(\rho)}{\delta_{q+1}(\rho)} Q(k+q+1) \right]}{\ln \{ TQ(k+q+1) \}} \quad (110)$$

で与えられる。ここで

$$\delta_{q+1}(\rho) = \left[\frac{\rho P_{2q+1}(\rho, \gamma)}{P_{2q}(\rho, \gamma)} - 1 \right] / (\rho^2 - 1)$$

である。

$\rho(J)$ の評価には(103)式を用いる。ただし、 V は

$$V(k+q+2) \approx P_{2q}(\rho, \gamma) \quad (111)$$

を用いる。さらに、一定の評価値を用いるCCP反復の回数に次の制限を置く。

$$\begin{aligned} q_1 &\geq 10, & q_2 &\geq 10, & q_3 &\geq 14, & q_4 &\geq 20, \\ q_5 &\geq 20, & q_s &\geq 28 \quad (s \geq 6). \end{aligned} \quad (112)$$

CCPには二通りのアルゴリズムがある。

σ_1 - CCPアルゴリズム

- Gauss-Seidel 反復 ($\sigma_{q,1} = \sigma_{q,2} = 1$) を3回実行し、(109)式により $\rho^2(J)$ に対する評価値 γ_1^2 を求める。 $\gamma_1 > 0.9486$ ならば、 $\gamma_1 = 0.9486$ とする。
- γ_1 を用いて、CCP計算を10回行う。
- (110)式により C を計算する。 $C > 0.8$ ならば、CCPを繰返し、 $C < 0.8$ になったら、(d)に移る。
- 1回 Gauss-Seidel 反復を行う。(111)式と(103)式により、 γ_2 を求める。 γ_s が(105)式の条件に合わない時は、上限の値をとる。(b)以下の計算を繰返す。

σ_1 - RQ-CCPアルゴリズム

- Gauss-Seidel 反復を2回実行する。
- 次の式を用いて γ_1 を決める。

$$\rho_q^2(J) = \frac{[R_2(q)]^T B_2 D_1^{-1} B_1 R_2(q)}{[R_2(q)]^T D_2 R_2(q)} \quad (113)$$

CCP計算を5回行う。

- (110)式により、 C を計算し、 $C < 0.8$ になるまで、CCP計算を繰返す。 $C < 0.8$ になったら、(b)へ移る。

(80)式のところで、配列(ordering)に関する概念がでてきたが、配列によって収束性も変り得るため、Hageman は次の二通りの配列を数値実験でチェックしている。すなわち、ベクトル X の配列でいうと、

英国型配列(English ordering)は通常の配列で、

$$X^T = (X_1, X_2, \dots, X_{N-1}),$$

σ_1 - 配列(σ_1 - ordering)は

$$X^T = (\underbrace{X_1, X_3, \dots, X_2, X_4, \dots}_{\text{奇数番号のもの}}, \underbrace{\dots}_{\text{偶数番号のもの}}).$$

配列の問題に関してはYoung が精力的に研究を進めているが⁽⁵⁰⁾，ここではこの問題には立ち入らないことにする。

以上簡単にまとめを行った種々のアルゴリズムについてはHageman が1次元1群平板拡散問題で数値計算を行って比較評価している⁽⁵¹⁾。計算結果の一部として，Table 9に示した非均質問題の場合を紹介しておく。他の問題例の場合にも共通しているが，Table 10の(4)に見るように，加速因子として最適値を用いた場合では σ_1 - CCP法が一番収束が早い。また配列による収束の差はそれほど見られないが，2，3次元多群多領域問題では，配列による収束性への影響は未だ具体的にチェックされておらず，配列法は効果が薄いと直ちに結論を出すことは危険であろう。

(2)ウィーラント法の改良

先にも述べたように，ウィーラント法も緩和法と同じように，初期見積りが非常に重要である。ウィーラント法の改良を試みたのはFeldmark である⁽²⁰⁾。(47) ~ (50) 式のアルゴリズムで，基本モードの固有値 λ_1 とする。

$$\epsilon(n) = \frac{\lambda(n) - \lambda(n-1)}{\lambda(n)} \tag{114}$$

を定義し， λ_e' を

$$\frac{\lambda_e' - \lambda(n)}{\lambda(n)} > \sum_{i=n}^{\infty} \epsilon(i) \tag{115}$$

を満足するように決める。

$$C = A(n)$$

$$\frac{1}{A(n)} = \frac{1}{\lambda(n)} - \frac{1}{\lambda_e}$$

とすると，さらに $\lambda_e' > \lambda_1$ であるためには，

$$C < \frac{1 - \sum_{i=n}^{\infty} \epsilon(i)}{\sum_{i=n}^{\infty} \epsilon(i)} \tag{116}$$

であることが必要である。しかし，(116)式は面倒なので，簡単に

$$C < \frac{1}{\epsilon(n)} \tag{117}$$

としておく。Feldmark のアルゴリズムは次の通りである。

- (a) ϵ と λ_e を入力として与え、次の条件が満たされるようになるまで、 λ_e を用いたウィーラント反復計算を行う。

$$\left| \frac{\lambda(n) - \lambda(n-1)}{\lambda(n)} \right| < \epsilon$$

- (b)
$$\frac{1}{\frac{1}{\lambda(n)} - \frac{1}{\lambda_e}} < C < \frac{1}{\epsilon}$$

となるような C の値を求める。

- (c) λ_e' を

$$\lambda_e' = \frac{\lambda(n) \cdot C}{C - \lambda(n)}$$

によって求め、この λ_e' を用いてウィーラント反復を収束するまで行う。

通常のウィーラント法と Feldmark のアルゴリズムを 1 次元 10 領域 2 群拡散計算に用いた場合の CDC 3600 での計算時間を比較したのが Fig. 11 である (20)。通常のウィーラント法では、見積り値 λ_e によって計算時間が急激に変化するが、Feldmark のアルゴリズムでは λ_e に関係なく、計算時間はほぼ一定で、かつ加速性が非常によいことが分る。したがって、Feldmark のアルゴリズムを用いる場合、 λ_e の見積り精度にはこだわる必要はなく、入力データの作成は非常に楽になる。以上の議論から明らかなように、GAMBLE-5 のウィーラント法のルーチンを Feldmark のアルゴリズムで置き換えれば、GAMBLE-5 の収束性はさらに良くなるであろう。

Table 8. Number iterations required for convergence of the rebalance method as a function of $\chi^{(39)}$

Problem	χ					
	0.0	0.5	1.0	2.0	3.0	4.0
M1	32	11	8	11	14	16
M2	*	38	23	35	46	100
M3	32	11	8	12	16	19
M4	*	38	23	35	46	100
M5	37	11	8	7	8	8
M6	11	8	6	5	6	7

As for the definition of problems, see Table 5.

* Failed to converge.

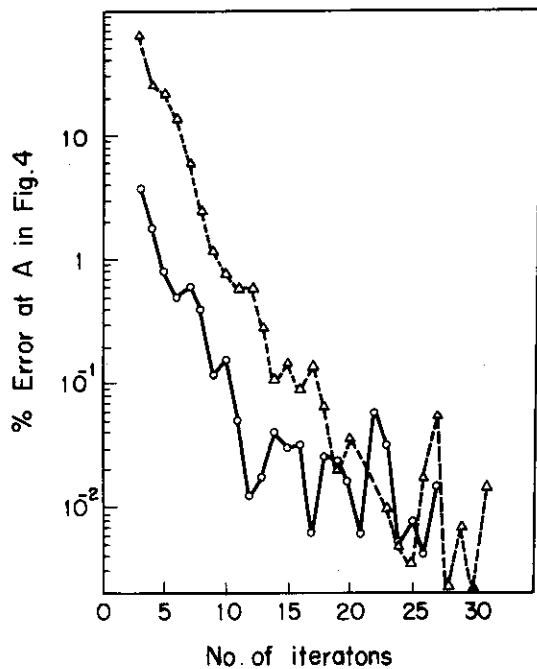


Fig. 9 Comparison of decay of flux errors at A in Fig.4 for space and space-angle coarse mesh rebalancing methods⁽⁵⁾

space coarse mesh rebalancing (6 x 6)
space-angle coarse mesh rebalancing (6 x 6 x 4)

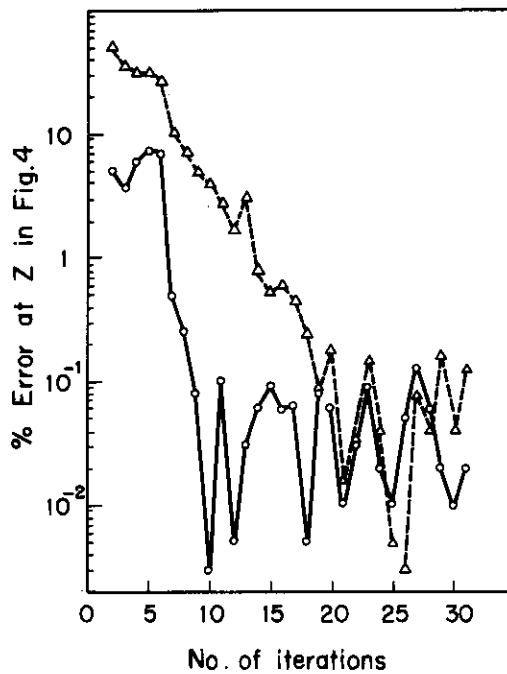


Fig. 10 Comparison of decay of flux errors at Z in Fig.4 for space and space-angle coarse mesh rebalancing methods⁽⁵⁾

space coarse mesh rebalancing (6 x 6)
space-angle coarse mesh rebalancing (6 x 6 x 4)

Table 9. Model problem characteristics for the evaluation of effectiveness of several SOR algorithms⁽²³⁾

One dimensional slab with $\Delta x = 0.1$, No. of meshes = 60		
Region	D(x)	$\Sigma(x)$
$0 \leq x \leq 0.8$	0.50	0.10
$0.8 \leq x \leq 2.2$	1.60	0.00
$2.2 \leq x \leq 2.6$	0.23	0.01
$2.6 \leq x \leq 3.8$	1.70	0.00
$3.8 \leq x \leq 4.6$	0.40	0.07
$4.6 \leq x \leq 6.0$	3.00	0.00

Table 10. Comparison of several SOR algorithms in a one dimensional diffusion problem⁽²³⁾

	Algorithm	No. of iterations		Last estimate of $\rho^2(j)$
		$\delta=10^{-2}$	$\delta=10^{-5}$	
1	Carré (σ_1 order)	138	216	0.998165
	Reid (English order)	155	251	0.997982
	Reid (σ_1 order)	124	209	0.998004
	σ_1 - RQ	120	205	0.998003
2	σ_1 - CCP	107	186	0.997998
	σ_1 - RQ - CCP	112	199	0.998003
3	CP (English order)	211	367	0.998004
4	SOR with exact ρ_b	79	164	exact ρ^2
	CP with exact $\rho(J)$	66	145	= 0.998004

δ stands for flux convergence criterion.
The flat initial guess was used for flux.

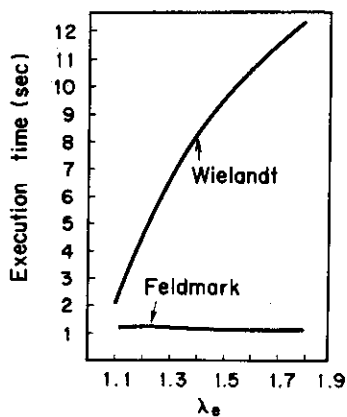


Fig.11

Execution time (CDC-3600) as a function of estimated eigenvalue for Wielandt and Feldmark algorithms in one dimensional two group diffusion problem

Flux and eigenvalue convergence criteria are 10^{-5} and 10^{-6} , respectively.

5. ま と め

一般に流布している S_N コード、拡散コードに用いられている収束法のまとめ、数値実験例において各々の加速法の評価を行い、これらコードをさらに改良して行く上での指針を示した。ここで、種々のコードの改良できる点をまとめておこう。

- (1) 1次元 S_N コードANISN及びDTF-IVについては
 - (a) エネルギー及び空間についての粗メッシュ再釣合い法を導入する。
 - (b) 詳細メッシュ上の反復計算においては、内側反復の加速にCCP法を、外側反復の加速にチェビシェフ加速法を用いるようにする。
- (2) 2次元 S_N コードTWOTRANについては、エネルギーについての粗メッシュ再釣合い法を導入する。DOTについては、エネルギー及び空間についての粗メッシュ再釣合い法を導入する。
- (3) 2次元拡散コードGAMBLE-5については、ウィーラント法をFeldmarkのアルゴリズムに置き換える。EXTERMINATORについては
 - (a) エネルギー及び空間についての粗メッシュ再釣合い法を導入する。
 - (b) 緩和法はCCP法に置き換える。

勿論今後議論してきた加速法が全てではなく、さらに研究を進めれば、より効果的な加速法を考え出すこともできよう。また、既に提案されている加速法でも、未だ核計算コードで試みられていないものについては積極的にこれの応用を試みても重要であろう。

この報告書では差分アルゴリズムにのみ議論の対象を限定してきたので、加速法という考え方とは別に解法そのものの工夫によって計算の能率化を計る他の試み、たとえば、有限要素法(finite element method)、合成法(synthesis method)等はとり上げなかった。また3次元コードにも言及しなかったが、その理由は3次元コードでは2次元コードとは基本的に異った新しいアルゴリズムが必要であり、2次元コードで用いられているアルゴリズムを単に延長しただけでは、実用的なコードは作れないだろうという判断によるものである。

また収束誤差の評価に関連した誤差解析理論には全く触れなかった。誤差解析はまだあまり開拓されていない研究分野であり、今後の研究が期待されるところである。

加速法について現在までに行われてきた数値実験例は、この報告書でも明らかなように、ほとんど簡単な1次元体系についてのみであり、2次元体系については数例をみるのみである。1次元計算で得られた結論をそのまま2次元問題に適用できない場合も多いことを考えると、今後特に2次元問題に重点を置いた系統的なベンチマーク・テストの実施が必要である。

なお本報告の執筆に当っては、できる限り数学的に統一された記述をするよう心掛けたが、場合によっては無理に統一すると、個々のコードで用いられている数学的手法について、そのマニュアルでの記述との関連が分りにくくなる場合もあるので、適宜表現を変えた部分もある。

文 献

- (1) Varga R. S. : "Matrix Iterative Method", Prentice Hall (1962)
- (2) Wachspress E. L. : "Iterative Solution of Elliptic Systems and Applications to the Neutron Diffusion Equations of Reactor Physics", Prentice Hall (1966)
- (3) Froehlich R. : "A Theoretical Foundation for Coarse Mesh Variational Techniques", Proc. Intern. Conf. Research Reactor Utilization and Reactor Mathematics, Mexico, D. F. 1, pp. 219-237 (1967)
- (4) Nakamura S. :
Nucl. Sci. Eng. , 39, 278 (1970), *ibid.*, 43, 116 (1971)
- (5) Nakamura S. : "New Formulation and Coarse-Mesh Acceleration for Two-Dimensional DS_N and P_N Methods", Proceeding of a Seminar on Numerical Reactor Calculations, IAEA, Vienna, pp. 51-82 (1972)
- (6) Dorsey J. P., Froehlich R. : "GAMBLE-5, A Program for the Solution of the Multigroup Neutron-Diffusion Equations in Two Dimensions, with Arbitrary Group Scattering, for the UNIVAC-1108 Computer", GA-8188 (1967)
- (7) Lathrop K. D., Brinkley F. W. : "Theory and Use of the General-Geometry TWOTRAN Program", LA-4432 (1970)
- (8) Froehlich R. : "Flux Synthesis Methods versus Difference Approximation Methods for the Efficient Determination of Neutron Flux Distributions in Fast and Thermal Reactors", Proceeding of a Seminar on Numerical Reactor Calculations, IAEA, Vienna, pp. 591-611 (1972)
- (9) Carlson B. G., Lathrop K. D., "Transport Theory, The Method of Discrete Ordinates", Computing Methods in Reactor Physics, ed. H. Greenspan, C. N. Kelber and D. Okrent, Gordon and Breach, pp. 171-266 (1968)
- (10) Dorsey J. P. : "GAMBLE-4. A Program for the Solution of the Multi-group Neutron-Diffusion Equations in Two Dimensions, with Arbitrary Group Scattering for the UNIVAC-1108 Computer", GA-6540 (1965)
- (11) Fowler T. B., Tobias M. L., Vondy D. R. : "EXTERMINATOR-II. A FORTRAN-IV Code for Solving Multigroup Neutron Diffusion Equations in Two Dimensions.", ORNL-4078 (1967)
- (12) RSIC Computer Code Collection, "ANISN, A One-dimensional Discrete Ordinates Transport Code", CCC-82 (EW220(1))

- (13) Lathrop K.D. : "DTF-IV, A FORTRAN-IV Program for Solving the Multigroup Transport Equation with Anisotropic Scattering", LA-3373 (1965)
- (14) RSIC Code Collection, "DOT, Two Dimensional Discrete Ordinates Transport Code", CCC-69 (K-1694)
- (15) 文献(2) 第6章
- (16) Fowler T.B., Vondy D.R. : "Nuclear Reactor Core Analysis Code: CITATION", ORNL-TM-2496 (1969)
- (17) Reed W.H., Nucl. Sci. Eng., 45, 245 (1971)
- (18)* Archibald R, Lathrop K.D., Mathews D. : "1DFX. A Revised Version of the 1DF (DTF-IV) S_n Transport Theory Code, GA-B-10820 (1971)
- (19) Renken J.H., Adams K.G. : "An Improved Capability for Solution of Photon Transport Problems by the Method of Discrete Coordinates", SC-RR-69-739 (1969)
- (20) Feldmark G.E. : "A Method for the Numerical Solution of Eigenvalue Problems in Diffusion Theory", Proceeding of a Seminar on Numerical Reactor Calculations, IAEA, Vienna, pp. 497-507 (1972)
- (21) 文献(2), 第9章
- (22) 文献(2), 第4章
- (23) Hagemann L.A. : "The Estimation of Acceleration Parameters for Chebyshev Polynomial and the Successive Over-relaxation Iteration Methods", WAPD-TM-1038 (1972)
- (24) Marlowe O.J., Saalbach C.P., Culpepper L.M., McCarty D.S. : "WANDA-A One-Dimensional Few Group Diffusion Equation Code for the IBM-704", WAPD-TM-28 (1956)
- (25) 鈴木友雄 : "高速炉用一次元拡散コードEXPANDA", JAERI-1091 (1965)
- (26) 伊勢武治, 久保淳洋 : "汎用一次元拡散コードGURNET", JAERI-1215 (1971)
- (27) Fowler T.B., Tobias M.L., Vondy D.R. : "EXTERMINATOR-A Multigroup Code for Solving Neutron Diffusion Equations in One and Two Dimensions", ORNL-TM-842 (1965)。計算法を見るには、このレポートを見た方がよい。ETXTERMINATOR-IIのマニュアルには計算式はほとんど書かれていない。
- (28) 文献(1), p. 123
- (29) 文献(2), 第6章

(*) このレポートは原研ではない。

- (30) Douglas J., Gunn J.: Numer. Math., 6, 428 (1964)
- (31) Reed W.H., Hansen K.F.: Nucl. Sci. Eng., 41, 431 (1970)
- (32) Wight A.L., Hansen K.F., Ferguson D.R.: Nucl. Sci. Eng., 44, 239 (1971)
- (33) Birkhofer A., Werner W.: "Fully Implicit Alternating Direction Method Applied to the Energy and Time Dependent Neutron Diffusion Equation", Proceeding of a Seminar on Numerical Reactor Calculations", IAEA, Vienna, pp. 689-696 (1972)
- (34) 文献(16) p. 730-10~730-11
- (35) 文献(6), p. 22~24
- (36) 文献(2), 第5章
- (37) Froehlich R.: "Benchmark Problem Solution (Benchmark problem ID. 3-A1)", Argonne Code Center Benchmark Problem Book, Benchmark Problem Committee of the ANS, ANL-7416 (1968), pp. 37-40
- このベンチマーク問題は均質2次元直方体で、6群定数を用いて、解析解と EXTERMINATOR-II, CITATION, GAMBLE-5 の結果を比較している。コードそのものの比較で、個々の加速法に注目した評価はしていない。
- (38) Reed W.H.: Nucl. Sci. Eng., 45, 245 (1971)
- (39) 文献(2), 第9章
- (40) 文献(2), 第8章
- (41) 文献(2), p. 264, Table 8, 7
- (42) Clancy B.E., Donnelly I.J.: Nucl. Sci. Eng., 39, 398 (1970)
- (43) 文献(1), p. 195
- (44) 文献(23), p. 13
- (45) 文献(23), p. 14
- (46) 文献(23), pp. 16~20
- (47) 文献(23), pp. 22~36
- (48) 文献(24), pp. 37~45
- (49) Yasinsky J.B., Natelson M., Hageman L.A.: "TWIGL-A Program to Solve the Two-Dimensional, Two-Group, Space-Time Neutron Diffusion Equations with Temperature Feedback", WAPD-TM-743 (1968)
- (50) Young D.M.: SIAM J. Numer. Anal., 9, 454 (1972), 及びこれに引用されている文献
- (51) 文献(23), pp. 46~75