

対称 3 重対角行列に対する固有値問題の L a x 方程式を用いた 並列アルゴリズムについて

電子技術総合研究所 関口智嗣 (SEKIGUCHI, Satoshi)
筑波大学情報学類 伊藤一宏 (ITOH, Kazuhiro)

1. はじめに

近年におけるベクトル計算機の普及はパイプライン処理というハードウェア技術を十二分に意識したアルゴリズムの開発に力を注ぐ契機となった。しかし同時にパイプライン処理の限界が認識されるようになってきた。アーキテクチャ技術は現在もパラレル（並列）処理というパイプライン処理とは異なったスーパーコンピュータを指向して様々な研究が続けられている。ところが並列処理を対象としたソフトウェア、とりわけ基本的な数値計算アルゴリズムに関してはまだ十分な研究がなされていないとは言い難い。並列アルゴリズムの開発には従来のアルゴリズムを改造して並列処理に適合させる方法と、全く独自に並列処理を対象としてアルゴリズムを開発する方法がある。本論文で取り上げる実対称 3 重対角行列の固有値計算も従来は 2 分法や QR 法が主流であり、並列アルゴリズムもこれらに基づいたもの (1) (2) があるだけであった。ここでは固有値問題を非線形連立常微分方程式に直して解くという新しいアルゴリズムについて紹介を行ない、逐次処理ではあるが多少の数値実験を行ったのでその結果を報告する。

2. 物理的背景

今回紹介するアルゴリズムのアイデアは1982年の「数学」12月号（岩波書店）にG. Strang教授講演記録（1984-07-14）として掲載されたものである。記法を代えてこれを紹介する。

L を対角成分が b_1, \dots, b_n ; その上下の並びが a_1, \dots, a_{n-1} ; その他が 0 の 3 重対角行列とする。

L の対角成分を 0 とし、その上の並びの符号を変えた交代行列を B とする。 a_j, b_j を t の関数とし

$$a_j(0) = a_j, \quad b_j(0) = b_j,$$

とするとき、 $L a x$ の方程式

$$\frac{dL(t)}{dt} = B(t)L(t) - L(t)B(t),$$

の解 $L(t)$ は、 $t \rightarrow \infty$ のとき L の固有値を成分とする対角行列に収束する。

$L a x$ 方程式の物理的な意味合いを付け加えておくと、これは指数型相互作用をもつ格子（戸田格子）の運動方程式を記述していることに相当している(3)。すなわち図1のような1次元格子において隣接した粒子が相互作用しているモデルを考える。

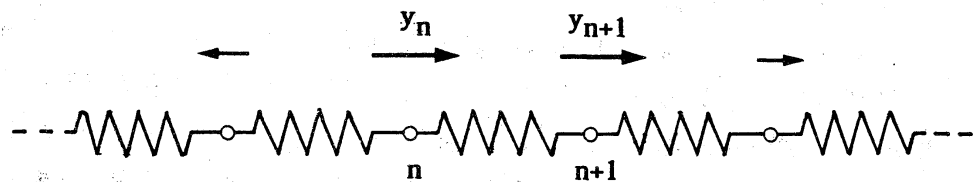


図1 1次元格子モデル

3. 数学的背景

$N \times N$ 実対称 3 重対角行列 $L = [l_{ij}]$ に対して行列 $B = [b_{ij}]$ 但し、

$$b_{ij} = l_{ij} \quad (i < j), \quad b_{ij} = 0 \quad (i = j), \quad b_{ij} = -a_{ij} \quad (i > j)$$

とする。 L, B を t の関数とするとき、

$$\frac{dL}{dt} = BL - LB$$

の解 $L(t)$ は (1) 固有値不変 であり、その固有値を $\lambda_1 < \lambda_2 < \dots < \lambda_N$

として (2) $\lim_{t \rightarrow \infty} L(t) = \Lambda$, したがって $\lim_{t \rightarrow \infty} B = 0$ となる。

[証明 (1)]

$L(t)$ に対してあるユニタリ変換 $U^t(t) L(t) U(t)$ を考える。

$$\begin{aligned} \frac{d}{dt}(U^t L U) &= \frac{dU^t}{dt} L U + U^t \frac{dL}{dt} U + U^t L \frac{dU}{dt} \\ &= \left(\frac{dU^t}{dt} + U^t B \right) L U + U^t L \left(\frac{dU}{dt} - B U \right) \end{aligned}$$

であるから、ここで U として

$\frac{dU}{dt} = B U$, $U^t(t_0) L(t_0) U(t_0) = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_N)$
を満たすものをとると、 $\frac{dU^t}{dt} = -U^t B$ より $\frac{d}{dt}(U^t L U) = 0$ で、

$$U^t L U = U^t(t_0) L(t_0) U(t_0) = \Lambda$$

となり、 $L(t)$ の固有値は $L(t_0)$ に等しいことがわかる。このような U は線形常微分方程式論における解の存在定理と一意性により存在があきらかである。

[注意] 以上の証明では L の対称性と B の反対称性しか用いていないので一般の対称行列とそれから構成される反対称行列の場合でも成立する。

[証明 (2)] <方針>

L の固有値 $\lambda_1, \lambda_2, \dots, \lambda_N$ に対応する固有ベクトルを $\mathcal{P}_1(t), \mathcal{P}_2(t), \dots, \mathcal{P}_N(t)$ とする。このとき

$$(L - \lambda I) \mathcal{X} = \mathcal{E}_1, \quad \mathcal{X} = (x_1, \dots, x_N)^t, \quad \mathcal{E}_1 = (1, 0, \dots, 0)^t \quad (3.1)$$

のような方程式を考えその解 \mathcal{X}_1 について2つの表現を与えることにより $a_n(t), b_n(t)$ を $\lambda_i, \mathcal{P}_j(t)$ を用いて表現する。

次に $a_n(t), b_n(t)$ を具体的に構成し $t \rightarrow \infty$ の状況を調べる。

[前半部]

一般に n 次正則行列 A の固有値を μ_1, \dots, μ_n に対応する固有ベクトルを v_1, \dots, v_n とすると $A\mathcal{X} = \mathcal{b}$ の解 \mathcal{X} は、

$$\mathcal{X} = \sum_j \frac{(b, v_j)}{\mu_j} v_j$$

で与えられる。式(3.1)にこれを適用すると λ が L の固有値と異なるとき正則となり、そのときの固有値は $\lambda_j - \lambda$ であるから

$$\mathcal{X}_1 = \sum_j \frac{\mathcal{P}_j^2(1, t)}{\lambda_j - \lambda}, \quad \mathcal{P}_j(t) = (\mathcal{P}_j(1, t), \dots, \mathcal{P}_j(N, t))^t$$

を得る。

$$\text{一方 } \Delta_{N-n+1} \equiv \begin{vmatrix} b_n - \lambda & a_n & \cdots & 0 \\ a_n & b_{n+1} - \lambda & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & a_{N-1} & b_N - \lambda \end{vmatrix}$$

とおくとき、クラメールの公式から

$$\mathcal{X}_1 = \frac{\Delta_{N-1}}{\Delta_N}$$

と表現できる。ここで Δ に関して成分を考えると

$$\Delta_{N-n+1} = (b_n - \lambda) \Delta_{N-n} - a_n^2 \Delta_{N-n-1}$$

という漸化式が成立している。したがって \mathcal{X}_1 を有限な連分数

$$\mathcal{X}_1 = \frac{\Delta_{N-1}}{\Delta_N} = \frac{1}{b_1 - \lambda} - \frac{a_1^2}{b_2 - \lambda} - \dots - \frac{a_{N-1}^2}{b_N - \lambda}$$

に展開することができ、 \mathcal{X}_1 に関して2とおりの表式を得る。

[注意] 第1の表現より有限の t について λ は $\lambda_1, \dots, \lambda_N$ を極として持つことから $a_1(t), \dots, a_N(t)$ は 0にならないことがわかる。

[後半部]

解の具体的構成はStieltjes (4)の方法を用いる。詳細な構成法は(5)に譲りここでは概略を示す。

まず各固有ベクトルの第1要素 $\varphi_j(t)$ は $L\varphi_j = \lambda_j \varphi_j$ と $\dot{\varphi}_j = B\varphi_j$ から具体的に積分できて

$$\varphi_j^2(t) = \frac{\varphi_j^2(1,0)e^{-2\lambda_j t}}{\sum_{i=1}^N \varphi_i^2(1,0)e^{-2\lambda_i t}} \rightarrow \varphi_j^2(t) > 0$$

が示される。次に z に関する部分分数式

$$f(z) = \sum_{i=1}^N \frac{\varphi_i^2}{\lambda_i + z}$$

はある定数 $\alpha_1, \dots, \alpha_{2N}$ を用いて

$$f(z) = \frac{\alpha_1 z^{-1}}{1} + \frac{\alpha_2 z^{-1}}{1} + \dots + \frac{\alpha_{2N} z^{-1}}{1}$$

と連分数表現できる。 α_i は $f(z)$ の $z = \infty$ における展開を

$$f(z) = \sum_{i=1}^N \frac{\varphi_i^2}{\lambda_i + z} = \frac{C_0}{z} - \frac{C_1}{z^2} \dots$$

$$C_j = \sum_{i=1}^N \lambda_i^j \varphi_i^2$$

とするときハンケル行列式

$$H_n^{(0)} = \begin{vmatrix} C_0 & C_1 & \dots & C_{n-1} \\ C_1 & C_2 & \dots & C_n \\ \vdots & \vdots & \ddots & \vdots \\ C_{n-1} & \dots & \dots & C_{2n-2} \end{vmatrix}, \quad H_n^{(1)} = \begin{vmatrix} C_1 & C_2 & \dots & C_n \\ C_2 & C_3 & \dots & C_{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ C_n & \dots & \dots & C_{2n-1} \end{vmatrix}$$

を用いて

$$\alpha_{2n+1} = \frac{H_{n+1}^{(0)} H_n^{(1)}}{H_n^{(0)} H_n^{(1)}}, \quad \alpha_{2n} = \frac{H_n^{(1)} H_{n-1}^{(0)}}{H_{n-1}^{(1)} H_n^{(0)}}$$

が成立する。

またこの構造より Binet-Cauchy の公式を適用して

$$H_k^{(0)} = \begin{cases} 0, & k > N \\ \sum_{n_1 < \dots < n_k} \left(\prod_{\substack{i < j \\ i, j \in \{n_1, \dots, n_k\}}} (\lambda_i - \lambda_j)^2 \varphi_{n_1}^2 \dots \varphi_{n_k}^2 \right), & k \leq N \end{cases}$$

$$H_k^{(1)} = \begin{cases} 0, & k > N \\ \sum_{n_1 < \dots < n_k} \left(\prod_{\substack{i < j \\ i, j \in \{n_1, \dots, n_k\}}} (\lambda_i - \lambda_j)^2 \lambda_{n_1} \dots \lambda_{n_k} \varphi_{n_1}^2 \dots \varphi_{n_k}^2 \right), & k \leq N \end{cases}$$

$\sum_{n_1 < \dots < n_k}$ は $1, 2, \dots, N$ の中から k 個の n_1, \dots, n_k を $n_1 < n_2 < \dots < n_k$ となるような取り出し方すべてについて加えることを意味する。

が得られる。したがって有限個の t を除いた全ての t に対して

$$H_1^{(0)} \neq 0, \dots, H_N^{(0)} \neq 0, H_{N+1}^{(0)} = 0, \dots$$

$$H_1^{(1)} \neq 0, \dots, H_N^{(1)} \neq 0, H_{N+1}^{(1)} = 0, \dots$$

となる。ここで

$$\frac{\alpha_1 z^{-1}}{1 + \frac{\alpha_2 z^{-1}}{1 + \dots \frac{\alpha_{2N-1}}{1 + \alpha_{2N} z^{-1}}}} = \frac{\alpha_1}{z + \frac{\alpha_2}{1 + \frac{\alpha_3}{z + \dots \frac{\alpha_{2N-1}}{z + \alpha_{2N}}}}}$$

さらに $z + \frac{\beta_1}{1 + \frac{\beta_2}{z}}$ に注意して

$$\frac{\alpha_1}{z + \frac{\alpha_2}{1 + \frac{\alpha_3}{z + \dots \frac{\alpha_{2N-1}}{z + \alpha_{2N}}}}} = \frac{\alpha_1}{z + \alpha_2 - \frac{\alpha_2 \alpha_3}{z + \alpha_3 + \alpha_4 - \frac{\alpha_4 \alpha_5}{z + \alpha_5 + \alpha_6 - \frac{\alpha_6 \alpha_7}{\dots \frac{\alpha_{2N-2} \alpha_{2N-1}}{z + \alpha_{2N-1} + \alpha_{2N}}}}}$$

となる。 $z = -\lambda$ と考えると

$$b_1(t) = \alpha_2, \quad a_1^2(t) = \alpha_2 \alpha_3,$$

$$b_2(t) = \alpha_3 + \alpha_4, \quad a_2^2(t) = \alpha_4 \alpha_5,$$

$$\vdots$$

$$b_n(t) = \alpha_{2n-1} + \alpha_{2n}, \quad a_n^2(t) = \alpha_{2n} \alpha_{2n+1}$$

$$\vdots$$

が得られ解が構成できる。

以上の構成から $t \rightarrow \infty$ のとき

$$b_j(t) \rightarrow \lambda_j, \quad a_j^2(t) \rightarrow 0$$

が示される。(証明終)

なお収束の証明だけならば微分方程式から簡単に示すことが可能である(5)。

4. 数値実験

L a x 形式にそのまま従う式 (2.1), (2.2) によると図 2 のような基本アルゴリズムが得られる。

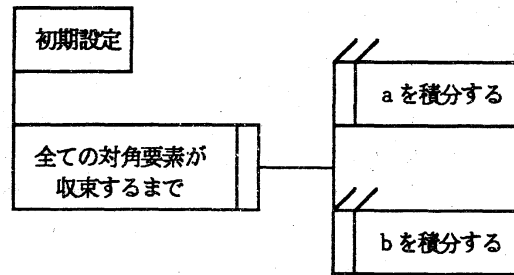


図 2 基本アルゴリズム

今回の実験では数値積分アルゴリズムとして 1 段 4 次のルンゲクッタ法を採用した。また収束列 $\{b_n\}$ に対して ϵ 加速を施してその効果を調べてみた。収束の判定条件はルンゲクッタの時間刻みを h として $mh \geq 1.0$,

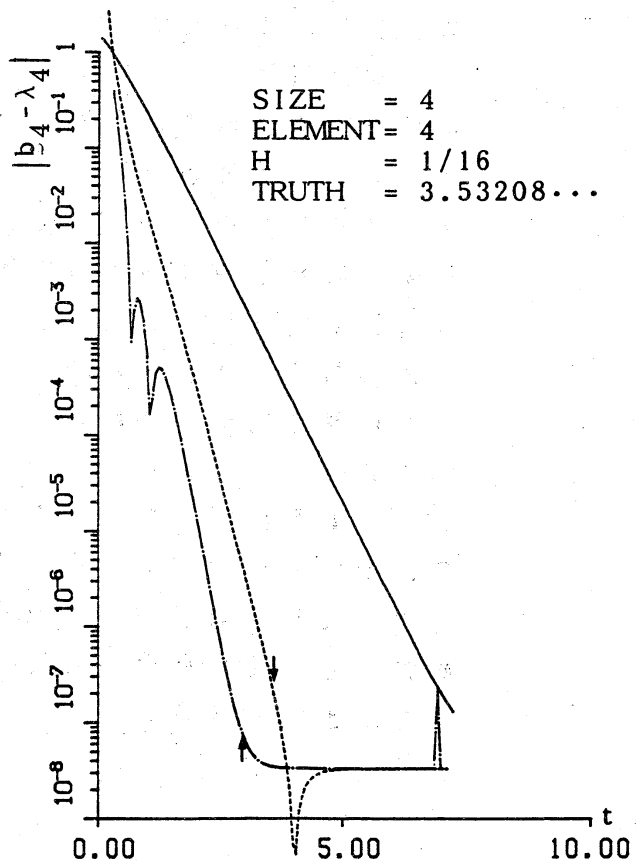
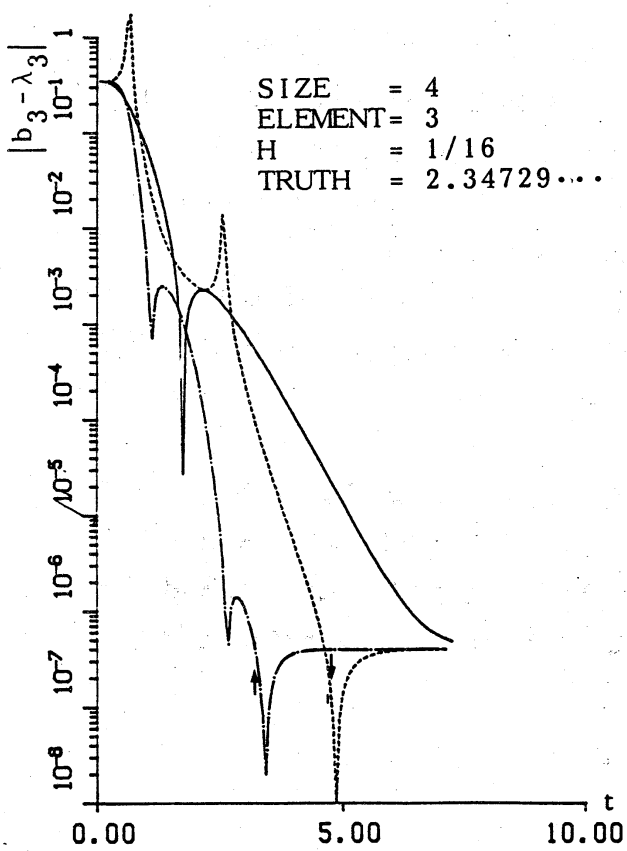
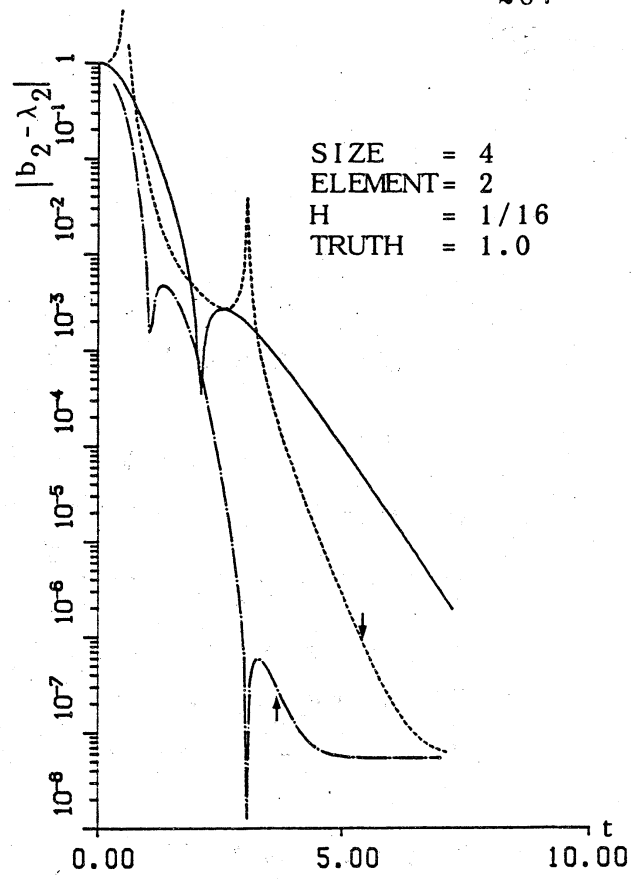
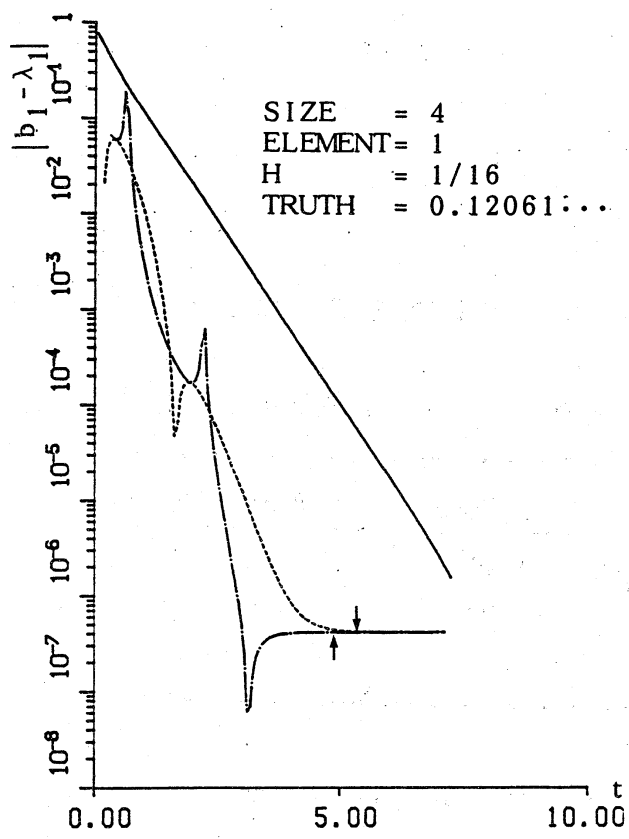
$$|b_j(t) - b_j(t-mh)| < \epsilon \quad \text{但し} \quad \epsilon = 10^{-5}$$

計算機は D G 社 MV4000 で Fortran77 倍精度を使用した。

テスト行列は解析的に固有値が得られるという理由から

$$T_n = \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & 0 \\ & -1 & 2 & \ddots & \\ 0 & & \ddots & \ddots & -1 \\ & & & -1 & 2 \end{pmatrix} \quad \lambda_i = 2 \left(1 - \cos \frac{2i-1}{2n+1} \pi \right)$$

を用いた。 T_4 の場合に対角要素の固有値への収束の様子を解析解との差で図 3 に示す。図中 SIZE はテスト行列の次数、ELEMENT は固有値番号、H はルンゲクッタの時間刻み、TRUTH は解析的に得られた固有値である。また実線はルンゲクッタ法、破線は ϵ 加速列の 1 段目、1 点鎖線は ϵ 加速列の 2 段目である。



↑ ϵ -加速の収束

図3 時間による対角要素の誤差の変動

これから収束までの時間とその時の誤差を表1にまとめた。

表1 収束までの時間と最良および最悪誤差 (SIZE=4,H=1/16)

	時間	最良誤差	最悪誤差
ルンゲクッタ法	7.25	1.1919×10^{-7}	1.9021×10^{-6}
ϵ 加速列の1段目	5.37	1.8366×10^{-7}	1.1162×10^{-6}
ϵ 加速列の2段目	3.62	8.4562×10^{-8}	3.3128×10^{-7}

固有値が十分分離されているテスト行列とはいえ誤差に関しては満足のいく結果が得られたといえる。また加速の効果も十分あることが示された。

次に同じテスト行列によるルンゲクッタの刻み幅と誤差の関係を図4に示す。いずれの場合も誤差は刻み幅のほぼ4乗のオーダーという理論値を保っているといえる。しかしルンゲクッタではその刻みを1/16以上細かくしても誤差は改善されないが、実際には収束列に関する情報は計算されているので加速により現出している様子を読み取れる。もっとも刻み幅を細かくすると丸め誤差の集積がおこり、加速を施すとさらに精度が低下するため理論値から外れ始める1/16~1/64あたりの刻み幅が安全であると考えられる。

計算量に関してテスト行列の次元 n と総演算量の関係を図5に示す。ルンゲクッタ法1回あたり $50n - 27$ の演算が必要であり、これに収束までの反復回数に乗じたものが総演算量で縦軸にとる。この傾きから総演算量は n のほぼ2.5乗に比例することが推定できる。

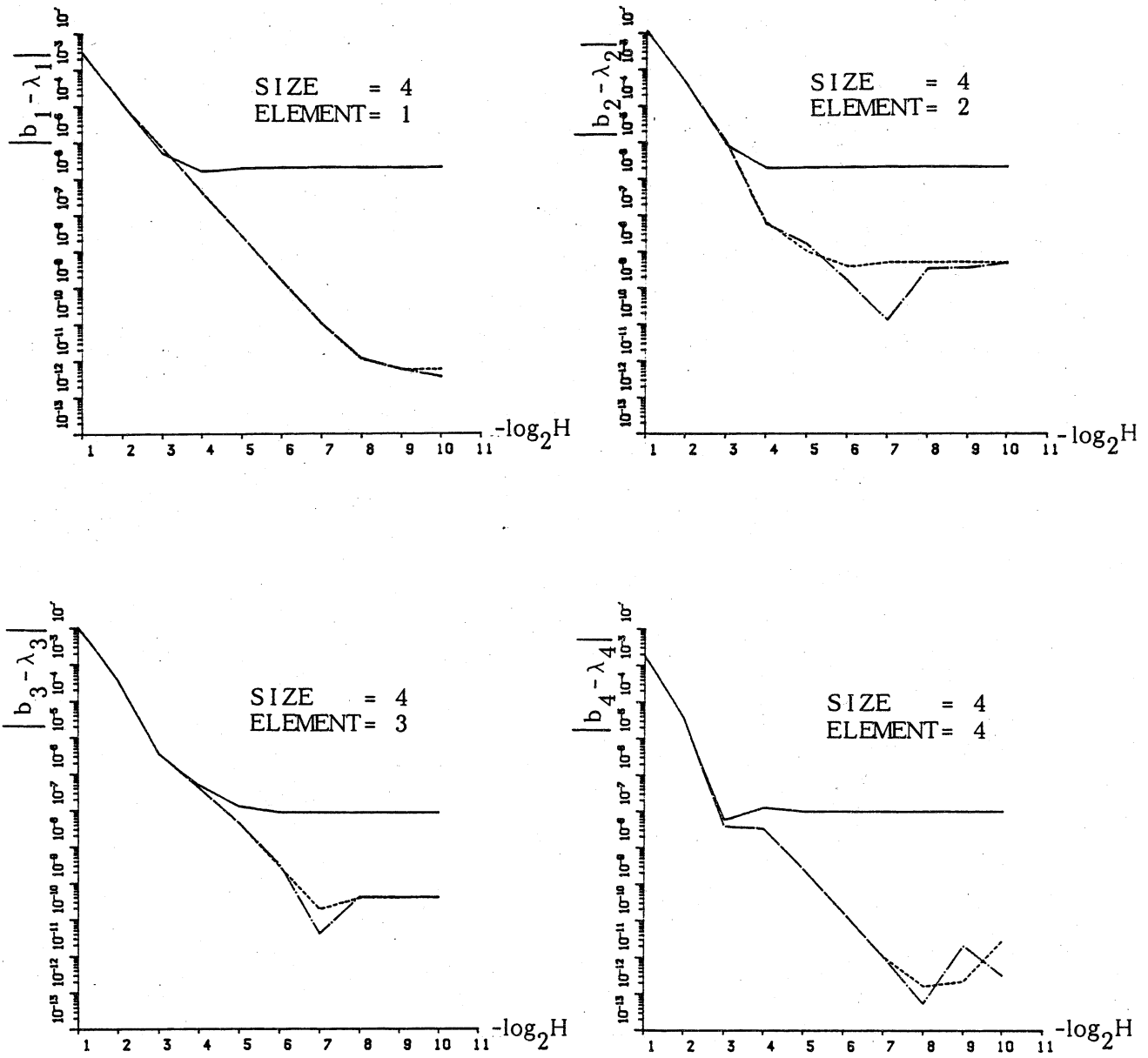


図4 ルンゲクッタの刻み幅と誤差の関係

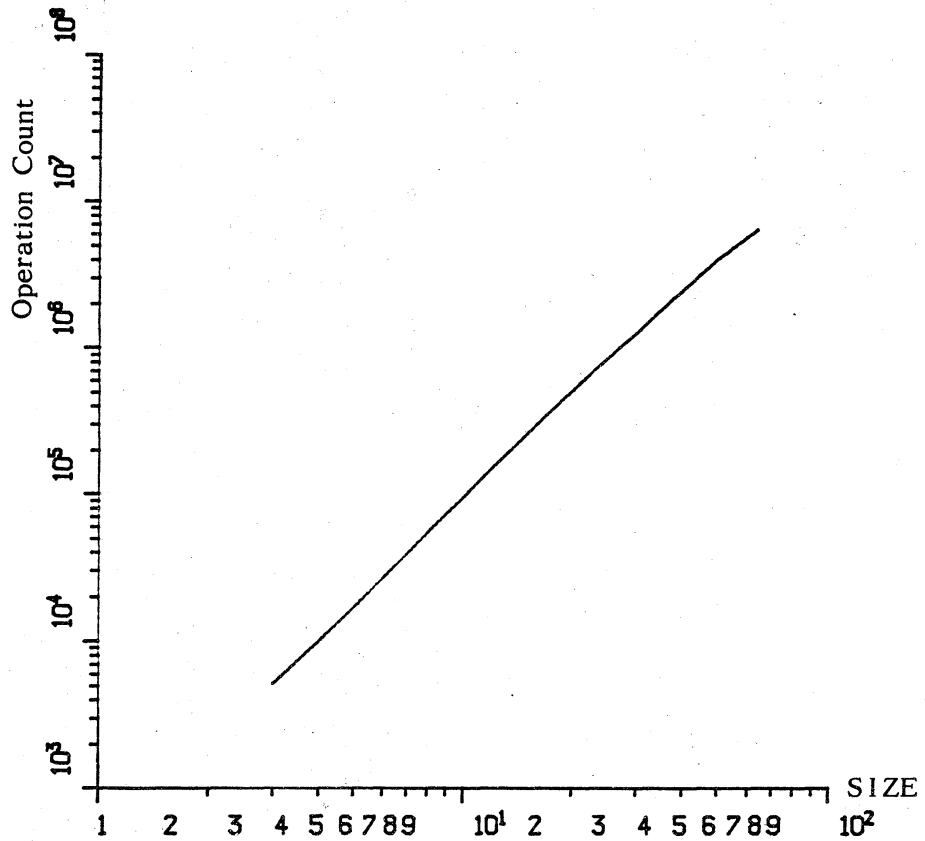
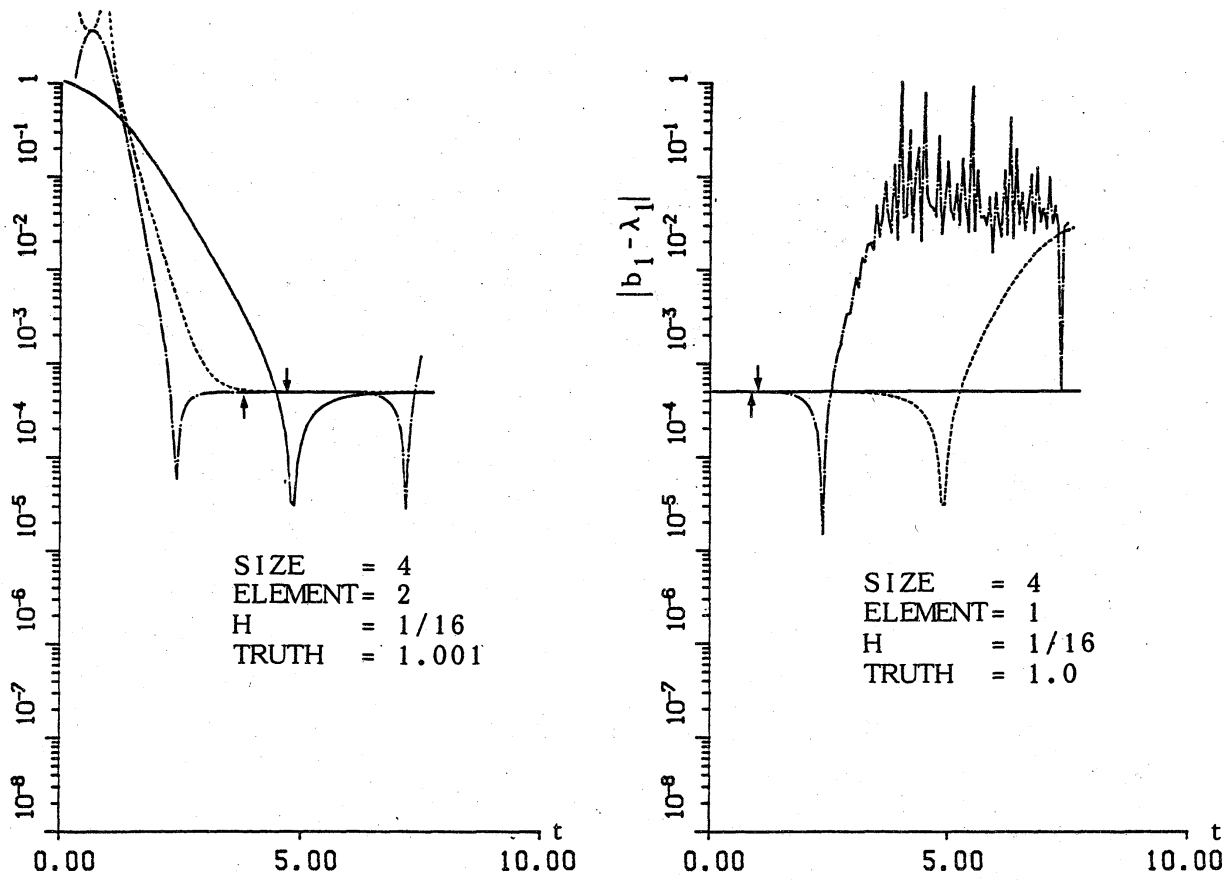


図5 テスト行列の次元 n と総演算量の関係

従来のアルゴリズムでも同様であるが固有値分布が近接している場合にはその分離が困難となる。このアルゴリズムでも解の構成からわかるように収束のオーダーが隣接固有値の差の指数乗である。したがって近接固有値が存在する場合には収束が非常に遅くなると考えられる。4次の例を作って実験を行ない、最も近接した1番、2番固有値について収束の様子を図6に示す。1番、2番固有値の差を ϵ とすると解の構成から収束の主要項は $\epsilon e^{-\epsilon t}$ となり、 ϵ のオーダーでしか分離できないことがわかる。図からもその状況が読み取れる。また1番固有値の例からもわかるように加速を施し過ぎたものは却て誤差が大きくなる。したがって反復の収束条件は加速を施した結果に対して判定すればよい。



$$\begin{array}{lll}
 b_1 = 1.000503, & a_1 = 0.001657, & \lambda_1 = 1.000000, \\
 b_2 = 2.090819, & a_2 = 0.514154, & \lambda_2 = 1.000999, \\
 b_3 = 2.159427, & a_3 = 0.828930, & \lambda_3 = 2.0, \\
 b_4 = 1.75025, & & \lambda_4 = 3.0,
 \end{array}$$

↑ ϵ -加速の収束

図6 近接固有値の収束の状況

5. まとめ

本論文ではLax方程式に基づく新しいアルゴリズムの背景を述べ、実際に数値実験を行なった結果を報告した。常微分方程式の積分をルンゲクッタ法で行ない満足のいく近似固有値

が得られ、 ϵ 加速が精度、速度ともに有効であることがわかった。また近接固有値を分離するには難点が残っている。

このアルゴリズムはその構成から常に n の並列性が存在している。ベクトル計算機、格子状結合並列計算機、データ駆動計算機のいずれにも適合することが予想される。実際は収束の速度が各要素によって異なるためベクトル長の短縮化がおきたり同期待ちの遊びが生じるため前 2 者は多少効率が低下することが考えられる。実装を行なって比較することには興味がある。このときには他の並列アルゴリズムとアーキテクチャに依存した比較が必要であろう。

熱心に御討論いただいた筑波大学電子情報工学系杉原正顕助手に感謝致します。

[参考文献]

- [1] Barlow, R. H. and Evans, D. J. "A Parallel Organisation of the Bisection algorithm," *Compt. J.* Vol. 22, No. 3, 1979, 267-269.
- [2] Sameh, A. H. and Kuck, D. J. "A Parallel QR Algorithm for Symmetric Tridiagonal Matrices," *IEEE Trans. Compt.* Vol. C-26, No. 2, 1977, 143-153.
- [3] 戸田盛和 非線形格子力学 岩波書店, 1978.
- [4] Henrici, P. "Applied and Computational Complex Analysis Vol. 2," Wiley, 1977.
- [5] 関口、伊藤、杉原：対称 3 重対角行列の固有値問題の $L a x$ 方程式を用いた解法、電総研情報部門瓦版 ETL-RM-86-3J.