4. 行列積の漸近的計算量†

野崎　昭弘††

1. 概　説

よく知られているように，n元連立1次方程式

$$Ax=b$$

は，係数行列 A が正則であれば，消去法で解くことができる．すなわち，“掃き出し”によって

$$Ex=c \quad (E \text{ は単位行列})$$

の形に変形でき，その際実行される4則演算は，乗除算も加減算も，どちらも約 $n^3/2$ 回である（ジョルダン法）．A の下半分だけをさきに掃き出し，代入法と組み合わせると，乗除算と加減算はどちらも約 $n^3/3$ 回ですむ（ガウス法）．

いっそうの高速化が可能であることを指摘したのはWinogradで，彼は1968年に次のことを証明した[16]．

（1）　ふたつの n 次正方行列の積は，約 $n^3/2$ 回の乗算と，約 $3n^2/2$ 回の加減算でできる．

（2）　その方法を応用すると，n 元連立1次方程式を，約 $n^3/6$ 回の乗算で解くことができる．

乗除算は（特に多倍長演算では）加減算より時間がかかるので，この方法によるある程度の高速化が期待できる．その実用的な価値はともかくとして，行列の積が線型計算の基本であって，連立1次方程式（や，実は逆行列）の計算にも深く関連していること，またこのような基本的な計算にも高速化の可能性があることを指摘した点で，意義が大きい．

これをさらにはっきりさせたのが，その翌年に出たStrassenの有名な論文[15]である．その中で彼は n 次正方行列の積や逆行列が，高々 $c\cdot n^{2.81}$ 回の4則演算で求められることを証明した．これは定数係数の改良でなく，次数の改良であるから，n が大きくなれば飛躍的な高速化が期待できる．このような改良は全く予想を越えるもので，大きな反響を呼んだ．

Strassen 以後，行列の積の計算量（演算回数）の研究は次のような方向で発展していった．

（1）　実用的な立場からの検討

（2）　他の線型計算との関連を調べる

（3）　よりよいアルゴリズムの開発——n 次正方行列の積を求めるのに十分な乗算回数を調べる．特に漸近的な計算量，すなわち n を大きくしていったときの演算回数の動向に興味がもたれている．

（4）　改良の限界の研究——n 次正方行列の積を求めるのに必要な乗算回数を調べる．

この小論では，まず(1)を簡単に紹介し，それからおもに(3)について，主要な結果を解説してみたい．

2. Strassen の方法とその実用法

Strassen の方法は次のふたつの着想に基づいている[1),2)]．

（1）　2次正方行列の積

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}=\begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

は，次のようにすれば，7回の乗算と18回の加減算で求めることができる．

$$m_1=(a_{12}-a_{22})(b_{21}+b_{22})$$
$$m_2=(a_{11}+a_{22})(b_{11}+b_{22})$$
$$m_3=(a_{11}-a_{21})(b_{11}+b_{12})$$
$$m_4=(a_{11}+a_{12})\cdot b_{22}$$
$$m_5=a_{11}\cdot(b_{12}-b_{22})$$
$$m_6=a_{22}\cdot(b_{21}-b_{11})$$
$$m_7=(a_{21}+a_{22})\cdot b_{11}$$
$$c_{11}=m_1+m_2-m_4+m_6$$
$$c_{12}=m_4+m_5$$
$$c_{21}=m_6+m_7$$
$$c_{22}=m_2-m_3+m_5-m_7$$

（2）　偶数次の正方行列の積は，各行列を次のように4等分することにより，2次の正方行列の積とみなすことができる．

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}\cdot\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}=\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

† Asymptotic Computational Complecity of matrix mnltiplication by Akihiro NOZAKI (International Christian University).

†† 国際基督教大学教養学部理学科

すると小行列の乗算7回，加減算18回で C_{ij} がすべて求められる．小行列が偶数次ならば，その積の計算にもこの方法が反復適用できる．

注意　小行列の積については，交換法則が成立しない．そこで反復適用を可能にするために，(1)では分配法則と結合法則しか利用していないことが本質的である．前章で述べた（乗算回数を約半分にする）Winograd の方法は，交換法則を利用しているので，(2)への応用はできない．

ここで理想的な場合，すなわち $n=2^p$ の場合の乗算回数を $M(p)$，加減算の回数を $A(p)$ であらわしてみよう．明らかに

$$M(1)=7,\quad A(1)=18$$

としてよい．また $p>1$ のとき，次の漸化式が成りたつ．

$$M(p)=7\cdot M(p-1)$$
$$A(p)=7\cdot A(p-1)+18\cdot(2^{p-1})^2$$

ここから単純な計算で，次の結果が得られる．

$$M(p)=7^p=n^{(\log_2 7)}\leqq n^{2.80735\cdots}$$
$$A(p)=6\cdot n^{\log_2 7}$$

n が 2^p の形をしていないときには，形式的に0を補って，2^p 次の正方行列として扱えば，やはり $O(n^{2.81})$ の乗算および加減算で積が求められる．なおふつうにやれば乗算 n^3 回，加減算 $n^2(n-1)$ 回である．(1)だけでは加減算の回数がひじょうにふえてしまうようであるが，反復適用によって，n を大きくしていけば**加減算の回数も少なくなる**点に注意しなければならない．

この方法について，実用的な見地から，次のような批判がある．

(ア)　c_{ij} を求めるのに，m_k の値を余分な項が打ち消されるように組み合わせている．逆にいえば「余分な項を加えてから引いている」わけで，有効数字最低1ビットの損はまぬがれがたい．これを p 回反復すれば，p ビットの損を覚悟しなければならない．

(イ)　プログラムが複雑になり，変数の添字処理の手間が大きくなる．また，再帰呼出しなどを利用すると，パラメータの受け渡しなどのオーバヘッドが大きく，処理時間はかえって大きくなる．

一般にプログラムの実行時間の中では添字処理の手間が無視できない割合を占めている．そのための演算回数を減らしても速くなるどころか，かえって遅くなることも考えられる．そのため「$n\leqq 512$ の範囲では，ふつうの方法の方が速い」とか，「アルゴリズムの差より，FORTRAN で書くかアセンブラで書くかの差の方が大きい」という報告もあるくらいである．

これらの批判に対しては，次のような提案がある．

(a)　特定の n に対しては，演算の順序と関連する変数の添字は確定しているはずで，プログラムが長くなることを厭わなければ，再帰呼出しや添字計算のないプログラムを書くことができる（吉田誠）．

(b)　方法(2)を徹底的に反復するのでなく，少数回の反復にとどめてふつうの方法に乗りかえる（佐藤中）．たとえば $2m$ 次正方行列の積に1回だけ方法(2)を適用し，小行列の積にはふつうの方法を適用すると，乗算回数は $7\cdot m^3$ 回，加減算の回数は

$$7\cdot m^2(m-1)+18\cdot m^2$$
$$=7\cdot m^3+11\cdot m^2$$

となる．これはふつうの方法（乗算 n^3 回，加減算 $n^2\cdot(n-1)$ 回）にくらべて，$n\geqq 32$ のときにすでに有利になる．

(c)　Winograd は Strassen の方法を改良して，2次正方行列の積を7回の乗算と15回の加減算で求められることを示した[1),2)]．

これによれば，$n=2^p$ の場合の加減算回数は約 $5\cdot 7^p$ となり，(b)と組み合わせれば中規模の問題にも十分有効になることが期待できる．

しかし残念ながら，よほど大きな n に対してでないときほどの効果が上げられないので，Strassen の方法の実用性については否定的な見方が一般的である．

3. 行列積の漸近的計算量

行列乗算の計算量について，理論的な興味は未だ尽きない．というよりはむしろ，ますますそのおもしろさが認識されてきた，といってもよい．その理由は，この問題が線型計算の計算量の基礎であること，きわめてむずかしい問題でその解決には深い理論が必要であろうと予想されること，そして実際，この問題をめぐっていくつかのおもしろい理論が提案されてきたこと，であろう．この小論でそのすべてを紹介することはもちろん不可能であるが，基本的な手法と成果を紹介するように努めてみたい．

以下，n 次正方行列の積を求めるのに必要十分な乗算回数を $M(n)$ であらわすことにしよう．また n 次正方行列の逆行列を求めるのに必要十分な乗算回数を $I(n)$，その行列式の値を求めるのに必要十分な乗算回数を $D(n)$ とする．さらに，n 元連立1次方程式を解くのに必要十分な乗算回数を $L(n)$ とすると，次の関

係が成りたつ.

$M(n) \leqq O(I(n))$,

$I(n) \leqq O(M(n))$,

$D(n) \leqq O(M(n))$,

$L(n) \leqq O(M(n))$.

証明は，たとえば Aho-Hopcroft-Ullman の教科書[1]の第6章に詳しく述べられている.

$M(n) \geqq n^2$

これも同書の第12章に述べられている（多項式の計算についての Winograd の理論 が役にたつ）.

$M(n)$ の上界については，Strassen の結果から

$M(n) \leqq O(n^{2.81})$.

また V. Pan によれば次の式が成りたつ[11].

$$M(n) \leqq O(n^{\alpha}), \qquad (\sharp)$$

ここで

$$\alpha = \log 143640/\log 70 \fallingdotseq 2.796.$$

その後，式(♯)をみたす次数 α の値は数年の間に急速に改良された．現在知られている最良の値は，Coppersmith と Winograd による 2.496 である[8].

ところで，計算すべき数式

$$z_{ij} = \sum_k x_{ik} y_{kj}$$

は双線型 (bilinear) である．これを計算するアルゴリズムは，その中の乗算がすべて

(x_{ij} の1次式)×(y_{ij} の1次式)

という形にあらわせるとき，**双線型アルゴリズム**と呼ばれる．行列の積を求める最良のアルゴリズムを考えるには，α の値についていえば，双線型アルゴリズムに限定して一般性を失わないことが知られている．詳しくいえば次のとおりである.

（1） 積の交換法則を使用しないアルゴリズムの乗算回数は，積の交換法則を使用したアルゴリズムの乗算回数の2倍以下におさえることができる[17].

（2） 積の交換法則を使用しないアルゴリズムの中で，双線型アルゴリズムが最小乗算回数を与える[5].

このようなことがあるので，双線型アルゴリズムは，Bini[3], Brockett[6], Chatelin[7] 等によって詳しく研究されている．さきに述べた α の値の改良にも，それらの理論が有効に使われている[10],[13].

次に α の値について，もう少し詳しく述べておこう.

$\alpha(n) = \log M(n)/\log n$,

$\alpha = \inf\{\alpha(n) \mid n > 1\}$

とおく．すると次のことがいえる.

事実 1[8]　任意の $m>1$ に対して，ある $n>m$ が存在して，$\alpha(n)<\alpha(m)$. すなわち，$\alpha=\alpha(m)$ をみたす m は存在しない.

事実 2[18]　ある m について $\alpha(m) \leqq c$ ならば，すべての n に対して $M(n) \leqq O(n^c)$.

事実 3[9]　$m \times n$ 行列と $n \times p$ 行列の積を求めるのに必要十分な乗算回数を $M(<m,n,p>)$ とする．そのときもし

$M(<m,n,p>) \leqq K$

ならば，

$\alpha \leqq 3\log K/\log mnp$

さて，これまでのアルゴリズムでは，ある体 F (ふつうは実数体) の上での計算を考えていた．Bini らはこれを拡張して，F に不定元をつけ加えた体 $F(\lambda)$ を考え，与えられた行列の成分からある有理式 $Z_{ij}(\lambda)$ を計算するアルゴリズムを考えた[4]．そして

$\lambda=0$ のとき $Z_{ij}(0)=z_{ij}$

となるならば，A は z_{ij} を**近似的に計算する**と呼ぶことにした．そこで $m \times n$ 行列と行列の積 $[z_{ij}]$ をこの意味で近似的に計算するのに必要十分な乗算回数を $M^*(m,n,p)$ であらわすと，次のことが成りたつ.

事実 4[4]　$\alpha = \inf\limits_{mnp>1} \{3\log M^*(m,n,p)/\log mnp\}$

Bini 等は $M^*(2,2,3) \leqq 10$ であることから，この事実によって $\alpha \leqq 2.779$ を導いた.

Schönhage はこれをさらに拡張して，$m_i \times n_i$ 行列と $n_i \times p_i$ 行列の積をすべて近似的に計算する ($1 \leqq i \leqq N$) のに必要十分な乗算回数を

$M^*(<m_1,n_1,p_1> \oplus \cdots \oplus <m_N,n_N,p_N>)$

であらわした．そして次のことを証明した.

事実 5[14]　$\prod m_i n_i p_i > 1$ で，しかも上記の量がある値 L でおさえられるならば $\alpha \leqq 3t$. ただし t は

$$\sum_{i=1}^{N} (m_i n_i p_i)^t = L$$

の解とする.

一方，ある簡単なアルゴリズムで

$M^*(<n,1,n> \oplus <1,(n-1)^2,1>) \leqq n^2+1$

を示すことができる．すると上の事実から，$n=4$ について $16^t+9^t=17$ を解き，$\alpha \leqq 2.54$ を導くことができる[14].

Coppersmith-Winograd の結果 $\alpha \leqq 2.496$ も，この方向の理論を精密化することによって導かれたものである.

4. 要　約

行列乗算の計算は双線型アルゴリズムによる近似計

算の概念によって，広く深く調べられるようになった．またここでは述べられなかったが三重線型形式(trilinear form)との関連やテンソル積への拡張も論じられている[12),18)]．しかしαの下界は2から少しも改良されておらず，上界との差はまだ大きい．理論の一層の進歩が期待されるわけである．また残念ながら，実用的アルゴリズムの開発からはしだいに離れてしまい，今のところ実際的な応用の可能性が乏しいことは，認めなければなるまい．

参 考 文 献

1) Aho, A. V., Hopcroft, J. E. and Ullman, J. D.: The Design and Analysis of Computer Algorithms, Addison-Wesley (1974) (邦訳：「アルゴリズムの設計と解析Ⅰ，Ⅱ」野崎・野下訳，サイエンス社)，第6章，第12章.
2) 伊理正夫，野崎昭弘，野下浩平：計算の効率化とその限界，日本評論社，第1章，第2章(1980).
3) Bini, D., Lotti, G. and Romani, F.: Suboptimal Solutions for the Bilinear Forms Computation Problem, Nota Interna, B 78-26, I. E. I. Pisa (1978).
4) Bini, D.: Relations between exact and approximate bilinear algorithms, Applications Calcolo, **17**, pp. 87-97 (1980).
5) Borodin, A. and Munro, L.: The Computational Complexity of Algebraic and Numeric Problems, American Elsevier (1975).
6) Brockett, R. and Dobkin, D.: On the optimal evaluation of a set of bilinear forms, Linear Algebra and its application, **19**, pp. 207-235 (1978).
7) Chatelin, P.: Une construction de l'algorithme de Winograd pour le produit de deux matrices 2×2, C. R. Acad. Sci. Paris, **290**. Série A, pp. 293-295 (1980).
8) Coppersmith, D. and Winograd, S.: On the Asymptotic Complexity of Matrix Multiplication, SIAM J. on Comp, **11**., pp. 472-492 (1982).
9) Hopcroft, J. E. and Musinski, J.: Duality applied to the complexity of matrix multiplication and other bilinear forms, SIAM J. on Comp., **1**, pp. 159-173 (1973).
10) Lazard, D.: Résultats recents sur le produit des matrices, Université de Poitiers (1980).
11) Pan, V. Y.: Strassen's algorithm is not optimal, Proc. 19th Annual Symp. on FCS, pp. 166-176 (1978).
12) Pan, V. Y.: An introduction to the trilinear technique, IBM Watson Research Center (1978).
13) Pan, V. Y.: New combinations of methods for the accelaration of matrix multiplication, Comp. Math., **7**, pp. 73-125 (1980).
14) Schönhage, A.: Partial and Total Matrix Multiplication, SIAM J. on Comp., **10**, pp. 434-455 (1981).
15) Strassen, V.: Gaussian Elimination is not optimal, Numerische Math., **13**, pp. 354-356 (1969).
16) Winograd, S.: A new algorithm for inner product, IEEE Trans. on Comp., pp. 693-694 (1968).
17) Winograd, S.: Arithmetic Complexity of Computation, Regional Conference Series in Applied Math., No. 33, SIAM (1980).
18) Winograd, S.: Asymptotic Complexity of Matrix Multiplication: Ideas and Results, the 7th IBM Symp. on MFCS at Hakone (1982).

(昭和58年2月15日受付)