Proof on Algorithm of Minimum Matrix Chain Calculation

Hiroki Shibata

2026, Jan.

行列 $M_1, M_2, ..., M_n$ があるとする。ここで、M_i は $p_{i-1} \times p_i$ の行列であるとする。$a \times b$ と $b \times c$ の 2 つの行列の積に計算に abc の計算量が必要だとする（ちなみにこれより少なく済む奇跡のような 2 つの行列積の計算法が存在するらしいが、ここでは省略する。)。この時、以下の行列の積、

$$M_i M_2 \cdots M_j$$

を計算するための最小の計算量を $f(i,j)$ とする。この時、$f(1,n)$ を $O(n^3)$ で求めるアルゴリズムが存在する。このことを数学的帰納法により証明する。

$n=1$ の時は自明である。自然数 k について $n=k$ の時、任意の $i,j(0<i\leq j\leq n)$ についての $f(i,j)$ のすべてを、ある正の実数 A に対して高々 Ak^3 で求めることが可能なアルゴリズムがあるとする（仮定 1)。つまり、$f(i,j)$ を求めるのに必要な計算量を $g(i,j)$ とした時、

$$\sum_{j=1}^{k} \sum_{a=1}^{j} g(i,j) < Ak^3$$

が成り立つとする（上記は $g(i,j)$ を i,j について独立に求めるという見方によれば矛盾している式に見えるが、メモ化を行うと考えると、i,j の幾らかについて計算量が減るため、成立させることが可能となる。)。

$n=k+1$ の時、$n \leq k$ の仮定の中でまだもとまっていない計算量は $i=1,...,k+1$ についての $f(i,k+1)$ である。これを求める。

ここで、別の帰納法により、仮定 1 が成り立つ上で、$f(i,k+1)$ を $i=k+1, k..., k+1-q$ のすべてについて求めるための計算量が、ある正の実数 C に対して高々 Cq^2 であることを証明する。これは $q=0$ の時は自明である。$q=l$ の時、成り立つとすると、$q=l+1$ の時、追加で必要な要素は、$f(k-l,k+1)$ である。これを求めるための行列積の結合が、以下、

$$(M_{k-l} \cdots M_j)\,(M_{j+1} \cdots M_{k+1})$$

について考えられるから、$f(k-l,k+1)$ の最小性の定義より、

$$f(k-l,k+1) = \min_{j=k-l,...,k} (f(k-l,j) + f(j+1,k+1) + p_0 p_i p_{k+1})$$

である。上記の最小値の各候補の計算には、$f(k-l,j), f(j+1,k+1), p_0 p_i p_{k+1}$ の三つの項が必要である。これについて、仮定より $f(k-l,j)$ はすでにもとまっており計算が追加で必要なく、$f(j+1,k+1)$ は $j+1 \geq l$ でありこれは仮定を満たすのですでにもとまっている。最後の $p_0 p_i p_{k+1}$ は冒頭の 2 つの行列積

の計算量の定義である。よって、$f(k-l,k+1)$ に必要な追加の計算量は、上記 3 つの項間に適用する算術に必要十分な計算量をある正の実数 D であるとすれば、これを $k-(k-l)+1=l+1$ 回繰り返すので、$D(l+1)$ である。合計すると、$D \leq C$ と取れば、

$$Cl^2+D(l+1)<Cl^2+C(l+1)+Cl=C(l+1)^2$$

が成り立つから、帰納法により $f(i,k+1)$ を $i=k+1,...,q$ のすべてについて求めるための計算量は高々 Cq^2 である。

　k の帰納法に戻れば、仮定 1 が成り立つ上で、つまり任意の $i,j(0<i\leq j\leq n)$ についての $f(i,j)$ のすべてを、ある正の実数 A に対して高々 Ak^3 で求めることが可能なとき、$f(i,k+1)$ をすべての $i=k+1,k,...,1=k+1-k$ について求めるための計算量は、今示したように、高々 Ck^2 であるから、仮定の計算量を足すと、$C\leq 3A$ となるように A を取れば、

$$Ak^3+Ck^2<A(k+1)^3=Ak^3+3Ak^2+3Ak+A$$

が成り立つから、$i,j(0<i\leq j\leq n)$ についての $f(i,j)$ のすべてを求めるための計算量は高々 $A(k+1)^3$ である。よって、数学的帰納法により、これがすべての n について成り立つから、$i=1,j=n$ を代入すれば、$f(1,n)$ を求めるための計算量が高々 An^3 であるアルゴリズムの存在を示せる。（証明終わり）

1　終わり

　なぜ An^3 と予想がつくか、それは 3 重ループのアルゴリズムが実際にあるからである。なぜ途中 Cn^2 で増加分がもとまるとわかるか、それは n^3 の階差数列が n^2 のオーダーであるからである。天下り式であることは、帰納法を使う上で仕方がないが、発想が正しいことを証明することに帰納法の良さがあるのである。

　難しかったです。ここの証明が正しいことの自信すらないので、よく検証してください。