Álgebra Lineal

Tema 1. Multiplicación rápida de matrices

Grado en Ingeniería Informática

Doble Grado en Ingeniería Informática y Administración de Empresas

AUTORES: J. SALAS, A. TORRENTE Y E.J.S. VILLASEÑOR

uc3m | Universidad Carlos III de Madrid

Índice general

1. Multiplicación rápida de matrices **1**

1.1. Algoritmo de Strassen (1969) 2

Tema 1

Multiplicación rápida de matrices

En el Tema 1 hemos visto cómo se define la multiplicación de matrices. En particular, si tenemos dos matrices A, B reales de dimensión $n \times n$ tenemos que efectuar n^3 multiplicaciones y $n^2(n-1)$ sumas. Si asumimos que las multiplicaciones de números de coma flotante son las operaciones más costosas de todo el proceso, entonces el tiempo que se tarda en calcular el producto de dos matrices de dimensión $n \times n$ es, cuando $n \gg 1$, proporcional al número de multiplicaciones M. En este caso, este número se comporta cuando $n \gg 1$ como $M \approx A\,n^3$, donde A es una cierta constante y estamos ignorando todas las correcciones (debidas a los otros tipos de operaciones) que se hacen cada vez más pequeñas conforme aumenta n.

El que el número de operaciones de este algoritmo para multiplicar dos matrices se comporte, cuando $n \gg 1$, como n^3 implica que éste es un problema "tratable" (en general basta con que dicho número de operaciones crezca como un polinomio en el "tamaño" n del sistema que queremos resolver).

Sin embargo si n es muy grande (del orden $n \gtrsim 10^6$) el número de operaciones será también muy grande. Para reducirlo se pueden intentar dos métodos:

- Disminuir la constante A.

- Disminuir la potencia, de manera que $M \approx B\, n^{\delta}$ con $\delta < 3$ cuando $n \gg 1$.

Desde luego que el segundo método es mucho más eficiente que el primero, pero a veces, reducir la potencia implica aumentar la constante $B \gg A$ y eso hace que, en la práctica, la ganancia no sea tan buena. De hecho para observar una disminución apreciable del número de operaciones hay que considerar valores de n demasiado grandes comparados con los tamaños típicos del problema en cuestión.

1.1. Algoritmo de Strassen (1969)

Vamos a empezar con la multiplicación de dos matrices 2×2: si $A\,B = C$, entonces

$$C = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = \begin{pmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{pmatrix}.$$

Es decir, necesitamos 8 multiplicaciones (y 4 sumas).

En el algoritmo de Strassen se calculan primero 7 variables intermedias m_i:

$$\begin{aligned}
m_1 &= (a_{11} + a_{22})\,(b_{11} + b_{22})\,, \\
m_2 &= (a_{21} + a_{22})\,b_{11}\,, \\
m_3 &= a_{11}\,(b_{12} - b_{22})\,, \\
m_4 &= a_{22}\,(b_{21} - b_{11})\,, \\
m_5 &= (a_{11} + a_{12})\,b_{22}\,, \\
m_6 &= (a_{21} - a_{11})\,(b_{11} + b_{12})\,, \\
m_7 &= (a_{12} - a_{22})\,(b_{21} + b_{22})\,.
\end{aligned}$$

En este paso se realizan 7 multiplicaciones y 10 sumas/restas. Finalmente los elementos de la matriz C se calculan a partir de los m_i como sigue

$$\begin{aligned}
c_{11} &= m_1 + m_4 - m_5 + m_7 \,, \\
c_{12} &= m_3 + m_5 \,, \\
c_{21} &= m_2 + m_4 \,, \\
c_{22} &= m_1 - m_2 + m_3 + m_6 \,.
\end{aligned}$$

En este segundo paso se necesitan 8 sumas/restas y ninguna multiplicación. Luego para calcular este producto de matrices se necesitan 7 multiplicaciones y 18 sumas.

En la práctica el algoritmo de Strassen sólo se puede usar con matrices de dimensión $N \times N$ con $N = 2^n$. Si A y B son matrices de dimensión $k \times k$ y k no es una potencia de 2, entonces ampliamos dichas matrices añadiendo ceros de manera que las nuevas matrices A', B' tengan dimensión $2^n \times 2^n$ donde n es el menor natural tal que $k < 2^n$. Se aplica ahora el algoritmo anterior de manera recursiva. Los elementos a_{ij}, b_{ij}, c_{ij} y m_i son ahora matrices de dimensión $2^{n-1} \times 2^{n-1}$ y la multiplicación se hace por bloques. Este algoritmo se repite n pasos hasta que los elementos que multipliquemos sean matrices 1×1, es decir números reales. La matriz C se obtiene quitando los ceros añadidos al principio. El número de multiplicaciones crece, cuando N es suficientemente grande, como $M \sim 7^n = 2^{n \log_2 7} = N^{\log_2 7} \approx N^{2{,}807}$, ya que $N = 2^n$. Esta es una mejora *marginal* con respecto al algoritmo original: $M \sim N^3$.

En la práctica las matrices A y B no necesitan ser cuadradas y la implementación se puede mejorar mucho. Sin embargo la memoria necesaria aumenta con respecto al algoritmo inicial, así como el número de otras operaciones. Aunque en un principio se pensaba que con este algoritmo sólo se obtenían ganancias significativas para matrices "muy grandes", en 2016 Huang, Smith, Henry y van de Geijn mostraron que se podía implementar de manera que fuera efectivo también para matrices "pequeñas".