빠른 3×3 행렬 곱셈법을 자동으로 찾아내기

Automatic Discovery of Fast 3×3 Matrix Multiplication

김진
최적화 및 금융공학 연구실
서울대학교

2011년 6월 26일

행렬 곱셈은 이런 문제들과 복잡도가 같습니다

- 역행렬
- 행렬식
- QR 분해
- 연립방정식 풀이

기본적인 행렬 곱셈

$$\begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{pmatrix} \times \begin{pmatrix} B_{11} & B_{12} & \cdots & B_{1n} \\ B_{21} & B_{22} & \cdots & B_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ B_{n1} & B_{n2} & \cdots & B_{nn} \end{pmatrix}$$

$$O(n^3)$$

2×2 행렬 곱셈

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \times \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$$

$$\begin{aligned} C_{11} &= A_{11} \times B_{11} + A_{12} \times B_{21} \\ C_{12} &= A_{11} \times B_{12} + A_{12} \times B_{22} \\ C_{21} &= A_{21} \times B_{11} + A_{22} \times B_{21} \\ C_{22} &= A_{21} \times B_{12} + A_{22} \times B_{22} \end{aligned}$$

곱셈 여덟 번 (2^3)

스트라센 알고리즘

$$
\begin{aligned}
M_1 &= (A_{11} + A_{22}) & \times \quad & (B_{11} + B_{22}) \\
M_2 &= (A_{21} + A_{22}) & \times \quad & B_{11} \\
M_3 &= A_{11} & \times \quad & (B_{12} - B_{22}) \\
M_4 &= A_{22} & \times \quad & (-B_{11} + B_{21}) \\
M_5 &= (A_{11} + A_{12}) & \times \quad & B_{22} \\
M_6 &= (-A_{11} + A_{21}) & \times \quad & (B_{11} + B_{12}) \\
M_7 &= (A_{12} + A_{22}) & \times \quad & (B_{21} + B_{22})
\end{aligned}
$$

$$
\begin{aligned}
C_{11} &= M_1 + M_4 - M_5 + M_7 \\
C_{12} &= M_3 + M_5 \\
C_{21} &= M_2 + M_4 \\
C_{22} &= M_1 - M_2 + M_3 + M_6
\end{aligned}
$$

곱셈 일곱 번

스트라센 알고리즘 분석

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \times \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

- 곱셈 일곱 번
- A_{ij}, B_{ij} 역시 행렬이라면?
- 재귀로 행렬 곱셈을 반복
- $T(n) = 7T(n/2) + O(n^2)$
- $T(n) = 7^{\log_2 n} = n^{\log_2 7} = O(n^{2.807\cdots})$

스트라센 이후

- $O(n^{2.376\cdots})$
  - 행렬 크기가 수백만에 달할 때 유효
- 2×2 행렬 곱셈은 일곱 번이 최적
- 2×2로 한정하지 않는다면?

스트라센 이후

- $O(n^{2.376\cdots})$
  - 행렬 크기가 수백만에 달할 때 유효
- 2×2 행렬 곱셈은 일곱 번이 최적
- 2×2로 한정하지 않는다면?

스트라센 이후

- $O(n^{2.376\cdots})$
  - 행렬 크기가 수백만에 달할 때 유효
- 2×2 행렬 곱셈은 일곱 번이 최적
- 2×2로 한정하지 않는다면?

스트라센 이후

- $O(n^{2.376\cdots})$
  - 행렬 크기가 수백만에 달할 때 유효
- 2×2 행렬 곱셈은 일곱 번이 최적
- 2×2로 한정하지 않는다면?

래더만 알고리즘

3×3 행렬 곱셈

$$
\begin{array}{lcl}
(A_{11}+A_{12}+A_{13}-A_{21}-A_{22}-A_{32}-A_{33}) & \times & B_{22} \\
(A_{11}-A_{21}) & \times & (-B_{12}+B_{22}) \\
A_{22} & \times & (B_{11}+B_{12}+B_{21}-B_{22}-B_{23}-B_{31}+B_{33}) \\
(-A_{11}+A_{21}+A_{22}) & \times & (B_{11}-B_{12}+B_{22}) \\
(A_{21}+A_{22}) & \times & (-B_{11}+B_{12}) \\
A_{11} & \times & B_{11} \\
(-A_{11}+A_{31}+A_{32}) & \times & (B_{11}-B_{13}+B_{23}) \\
(-A_{11}+A_{31}) & \times & (B_{13}-B_{23}) \\
(A_{31}+A_{32}) & \times & (-B_{11}+B_{13}) \\
(A_{11}+A_{12}+A_{13}-A_{22}-A_{23}-A_{31}-A_{32}) & \times & B_{23} \\
A_{32} & \times & (-B_{11}+B_{13}+B_{21}-B_{22}-B_{23}-B_{31}+B_{32}) \\
(-A_{13}+A_{32}+A_{33}) & \times & (B_{22}+B_{31}-B_{32}) \\
(A_{13}-A_{33}) & \times & (B_{22}-B_{32}) \\
A_{13} & \times & B_{31} \\
(A_{32}+A_{33}) & \times & (-B_{31}+B_{32}) \\
(-A_{13}+A_{22}+A_{32}) & \times & (B_{23}+B_{31}-B_{33}) \\
(A_{13}-A_{23}) & \times & (B_{23}-B_{33}) \\
(A_{22}+A_{23}) & \times & (-B_{31}+B_{33}) \\
A_{12} & \times & B_{21} \\
A_{23} & \times & B_{32} \\
A_{21} & \times & B_{13} \\
A_{31} & \times & B_{12} \\
A_{33} & \times & B_{33}
\end{array}
$$

곱셈 27번 $\rightarrow$ 23번

래더만 방법 응용

- 3×3 행렬 곱셈은 2×2 행렬 곱셈 여러 개로 이루어지고,
- 2×2 쌍마다 곱셈 수를 하나씩 줄인다 (스트라센 알고리즘)
- 2×2 네 개로 이루어진 래더만 알고리즘은 곱셈 23개
- 2×2 행렬 곱셈 다섯 개를 조합할 수 있다면?

$$\begin{pmatrix} \odot & \odot & \cdot \\ \odot & \odot & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix} \times \begin{pmatrix} \odot & \odot & \cdot \\ \odot & \odot & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix} \quad \begin{pmatrix} \odot & \odot & \cdot \\ \cdot & \cdot & \cdot \\ \odot & \odot & \cdot \end{pmatrix} \times \begin{pmatrix} \odot & \cdot & \odot \\ \odot & \cdot & \odot \\ \cdot & \cdot & \cdot \end{pmatrix}$$

$$\begin{pmatrix} \cdot & \odot & \odot \\ \cdot & \cdot & \cdot \\ \cdot & \odot & \odot \end{pmatrix} \times \begin{pmatrix} \cdot & \cdot & \cdot \\ \odot & \odot & \cdot \\ \odot & \odot & \cdot \end{pmatrix} \quad \begin{pmatrix} \cdot & \odot & \odot \\ \cdot & \odot & \odot \\ \cdot & \cdot & \cdot \end{pmatrix} \times \begin{pmatrix} \cdot & \cdot & \cdot \\ \odot & \cdot & \odot \\ \odot & \cdot & \odot \end{pmatrix}$$

최소 곱셈 횟수

n	하한	상한
2	7	7
3	18	23
4	33	49
5	52	100
6	75	161
7	102	258
8	136	343
9	176	522
10	220	700
11	270	923
12	324	1127
13	384	1450
14	448	1728
⋮	⋮	⋮