Module 8
Strassen’s Method

S. Lakshmivarahan
School of Computer Science
University of Oklahoma
USA-73019
Varahan@ou.edu

Strassen's Method for Matrix Multiplication
Divide and Conquer

- The conventional approach to multiplying two nxn matrices take n^3 multiplications and $n^2(n-1)$ additions
- Since it is well known that multiplication of two numbers takes longer time than addition of the same set of numbers, it behooves us to ask the following question

Strassen’s Method for Matrix Multiplication Divide and Conquer

- In multiplying to matrixes, can we trade the multiplications for additions so as to reduce the overall time required?
- This was answered affirmitavely by Strassen in 1969 in a short paper entitles “Gaussian Elimination is not optimal” that appeared in the journal Numerische Mathematik, vol 13, pages 354-356

Strassen’s Method for Matrix Multiplication Divide and Conquer

- We now introduce the basic ideas of Strassen’s algorithm
- Consider the multiplication of two 2x2 matrices using the classical method, consisting of the inner product of rows and columns

Strassen's Method for Matrix Multiplication Divide and Conquer

- $C = AB$
- $\begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$
- $c_{11} = a_{11}b_{11} + a_{12}\, b_{21} \rightarrow$ 2 multiply & 1 add
- $c_{12} = a_{11}b_{12} + a_{12}\, b_{22} \rightarrow$ 2 multiply & 1 add
- $c_{21} = a_{21}b_{11} + a_{22}\, b_{21} \rightarrow$ 2 multiply & 1 add
- $c_{22} = a_{21}b_{12} + a_{22}\, b_{22} \rightarrow$ 2 multiply & 1 add

Strassen’s Method for Matrix Multiplication
Divide and Conquer

- Thus, it takes 8 multiplications and 4 additions to compute the product, C=AB
- The basic idea of the Strassen’s algorithm is to rewrite these computations in such a way, it requires only 7 multiplications but 18 additions

Strassen’s Method for Matrix Multiplication
Divide and Conquer

- At first sight, it might seem that we are going in the wrong direction, but when applied recursively to large matrices with $n=2^k$, we will end up saving the cost of multiplying the sequences of matrices of sizes $\frac{n}{2}, \frac{n}{2^2}, \frac{n}{2^3}, \ldots, \frac{n}{2^k}$.
- The net saving would far exceed the additional matrix additions resulting in an algorithm that takes $O(n^{2.81})$ instead of the $O(n^3)$ time.

Strassen’s Algorithm

- The following is Strassen’s algorithm for multiplying 2x2 matrices A and B

Strassen's Algorithm

- **Step 1**: First, compute a set of seven intermediate quantities x_1, x_2, … , x_7, requiring 7 multiplications and 10 additions, as follows: (m = multiply, a = add)
  - $X_1 = (a_{11} + a_{22}) * (b_{11} + b_{22})$ → 1-m, 2-a
  - $X_2 = (a_{21} + a_{22}) * b_{11}$ → 1-m, 2-a
  - $X_3 = a_{11} * (b_{12} - b_{22})$ → 1-m, 2-a
  - $X_4 = a_{22} * (b_{21} - b_{11})$ → 1-m, 2-a
  - $X_5 = (a_{11} + a_{12}) * b_{22}$ → 1-m, 2-a
  - $X_6 = (a_{21} - a_{11}) * (b_{11} + b_{12})$ → 1-m, 2-a
  - $X_7 = (a_{12} - a_{22}) * (b_{21} + b_{22})$ → 1-m, 2-a

7-m, 10-a

1

Strassen's Algorithm

- **<u>Step 2</u>**: Now assemble the final results using only 8 additions
  - $C_{11} = x_1 + x_4 - x_5 + x_7$ → 3-a
  - $C_{12} = x_3 + x_5$ → 1-a
  - $C_{21} = x_2 + x_4$ → 1-a
  - $C_{11} = x_1 + x_3 - x_2 + x_6$ → 3-a

8-a

②

Strassen’s Algorithm

- We leave it as an exercise to check the correctness by reproducing the result obtained by the classical method
- Thus, together it requires 7 multiplications and 18 additions to multiply two 2x2 matrices

Strassen’s Algorithm

- Now, we extend Strassen’s algorithm to mutliplying two nxn matrices when n=2^k
- First, partition A and B into four sub-matrices each of size $\frac{n}{2} \text{x} \frac{n}{2}$ as follows:
- $\left[\begin{array}{c|c} c_{11} & c_{12} \\ \hline c_{21} & c_{22} \end{array}\right] = \left[\begin{array}{c|c} a_{11} & a_{12} \\ \hline a_{21} & a_{22} \end{array}\right] \left[\begin{array}{c|c} b_{11} & b_{12} \\ \hline b_{21} & b_{22} \end{array}\right]$

Strassen’s Algorithm

- Now, setting

$$a_{ij} \leftarrow A_{ij},\, b_{ij} \leftarrow B_{ij}, \quad \begin{matrix} 1 \le i \le 2 \\ 1 \le j \le 2 \end{matrix}$$

and using the same formula as Step 1 above, we can compute $\frac{n}{2} \text{x} \frac{n}{2}$ matrices x_i ($1 \le i \le 7$) that require 7 matrix multiplications and 10 matrix additions of $\frac{n}{2} \text{x} \frac{n}{2}$ matrices

Strassen's Algorithm

- **<u>Note</u>**: Each of the seven matrix multiplications of $\frac{n}{2} \times \frac{n}{2}$ matrices will be done recursively by partitioning each $\frac{n}{2} \times \frac{n}{2}$ matrix into four $\frac{n}{4} \times \frac{n}{4}$ submatrices.

Strassen's Algorithm

- Once the seven $\frac{n}{2} \times \frac{n}{2}$ matrixes x_i ($1 \leq i \leq 7$) are available, we can use Step 2 to assemble the four components c_{11}, c_{12}, c_{21}, c_{22} of the product matrix C

Complexity

- Recall matrix multiplication requires real multiplication and addition, while matrix addition requires only real addition
- We compute the number of multiplications and additions separately

Complexity
Total number of multiplications

- Let M(n) be the number of multiplications needed to multiply two nxn matrices with $n=2^k$.
- From Step 1 in ①, it follows

$$M(n) = 7M(n/2),\ n = 2^k$$

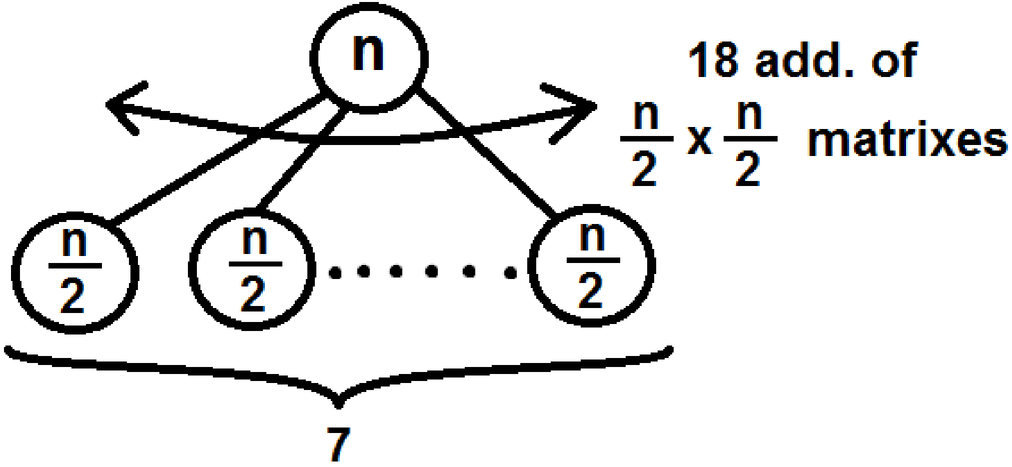
$$M(1) = 1$$

- Solving, it follows

$$M(n) = 7^{\log_2 n} = n^{\log_2 7} = n^{2.81}$$

Complexity
Total number of additions

- Let A(n) be the total number of additions in multiplying two nxn matrixes



Total number of additions

- $$A(n) = 7A\left(\frac{n}{2}\right) + 18\left(\frac{n}{2}\right)^2, n = 2^k \ \rightarrow (***)$$

Implicit additions hidden in the 7 recursive multiplicative calls

Explicit 18 additions of $\frac{n}{2}$x$\frac{n}{2}$ matrices in Steps 1 and 2

where

$$A(1) = 0$$

Total number of additions

- Iterating, we obtain: ($k = \log_2 n$ and $n = 2^k$)

- $A(n) = \sum_{i=0}^{k-1} 7^i * 18\left(\frac{n}{2^{i+1}}\right)^2$

$$= 18(2^k)^2 \sum_{i=0}^{k-1} \left(\frac{7}{4}\right)^i$$

$$= \frac{18}{4} n^2 \sum_{i=0}^{k-1} \left(\frac{7}{4}\right)^i$$

$$= \frac{9}{2} n^2 \frac{\left(\frac{7}{4}\right)^k - 1}{\left(\frac{7}{4}\right) - 1}$$

Total number of additions

$$= 6n^2\left[\frac{7^k}{4^k} - 1\right]$$

$$= 6 * 7^k - 6n^2 \quad (\because 4^k = 2^{2k} = n^2)$$

$$= 6 * 7^{\log_2 n} - 6n^2$$

$$= 6 * n^{\log_2 7} - 6n^2$$

$$= 6 * n^{2.81} - 6n^2$$

$$\therefore T(n) = M(n) + A(n) = 7n^{2.81} - 6n^2$$

$$= O(n^{2.81})$$

Extension to $n \neq 2^k$

- There are several ways to extend the above idea to the case when n is not a power of 2
- To illustrate, let n=11
- First, find the largest power of 2 less than n. In this case, $2^3 = 8 < 11$

Extension to n≠2^k

- Divide A,B as shown below

- $$\begin{array}{cc} & \begin{array}{cc} 8 & 3 \end{array} \\ \begin{array}{c} 8 \\ 3 \end{array} & \left[\begin{array}{c|c} c_{11} & c_{12} \\ \hline c_{21} & c_{22} \end{array}\right] \end{array} = \begin{array}{cc} & \begin{array}{cc} 8 & 3 \end{array} \\ \begin{array}{c} 8 \\ 3 \end{array} & \left[\begin{array}{c|c} a_{11} & a_{12} \\ \hline a_{21} & a_{22} \end{array}\right] \end{array} \begin{array}{cc} \begin{array}{cc} 8 & 3 \end{array} & \\ \left[\begin{array}{c|c} b_{11} & b_{12} \\ \hline b_{21} & b_{22} \end{array}\right] & \begin{array}{c} 8 \\ 3 \end{array} \end{array}$$

Extension to n≠2^k

- Then,

$$C_{11} = A_{11}B_{11} + A_{12}\ B_{21}$$

$$C_{12} = A_{11}B_{12} + A_{12}\ B_{22}$$

$$C_{21} = A_{21}B_{11} + A_{22}\ B_{21}$$

$$C_{22} = A_{21}B_{12} + A_{22}\ B_{22}$$

Extension to $n \neq 2^k$

- In here, A_{11} and B_{11} are of the size 2^k with k=3. Hence, this product can be obtained using the Strassen's method.
- The rest of the products can be obtained by using the classical method

Problems:

1) Verify the solution for the recurrences relating to the number of multiplications, M(n) and additions, A(n)

Extension to n≠2^k

- Note: Strassen's method triggered a stream of research into multiplying matrices faster
- A good summary of the results is contained in the monograph:
- V. Pan (1982) *How to multiply matrices faster*, Springer Verlag, New York,