



Name: Reza Shisheie

Login ID: reshishe

## 1. Exercise 4.2-6 (Textbook page 83):

Assuming

$$A_i = \begin{bmatrix} A_{i,1 \times 1} & \cdots & A_{i,1 \times n} \\ \vdots & \ddots & \vdots \\ A_{i,n \times 1} & \cdots & A_{i,n \times n} \end{bmatrix}_{n \times n}, B_i = \begin{bmatrix} B_{i,1 \times 1} & \cdots & B_{i,1 \times n} \\ \vdots & \ddots & \vdots \\ B_{i,n \times 1} & \cdots & B_{i,n \times n} \end{bmatrix}_{n \times n}, i \in [1, \dots, k]$$

where,

$$A = \begin{bmatrix} A_1 \\ \vdots \\ A_i \\ \vdots \\ A_k \end{bmatrix}_{kn \times n}, B = [B_1 \cdots B_i \cdots B_k]_{n \times kn}, i \in [1, \dots, k]$$

- Forward Solution:

Thus the product of  $A \times B$  is a  $kn \times kn$  matrix:

$$A \times B = \begin{bmatrix} A_1 \times B_1 & \cdots & A_1 \times B_i \\ \vdots & \ddots & \vdots \\ A_i \times B_1 & \cdots & A_i \times B_i \\ \vdots & \ddots & \vdots \\ A_k \times B_1 & \cdots & A_k \times B_i \end{bmatrix}_{kn \times kn}, i \in [1, \dots, k]$$

where time complexity of each  $A_i \times B_i$  costs  $\Theta(n^{\log_2 7})$  according to Strassen's algorithm. Since there are  $k^2$  matrix multiplications, time complexity of  $A \times B$  is:

$$T(n)_{A \times B} = k^2 \cdot \Theta(n^{\log_2 7}) = \Theta(k^2 \cdot n^{\log_2 7})$$

- Reverse Solution: For the reverse multiplication, the product of  $B \times A$  is a  $n \times n$  matrix:

$$B \times A = [B_1 \times A_1 + \cdots + B_i \times A_i]_{n \times n} = \left[ \sum_{i=1}^k B_i \times A_i \right]_{n \times n}$$

where time complexity of each  $B_i \times A_i$  costs  $\Theta(n^{\log_2 7})$  according to Strassen's algorithm. Since there are  $k$  matrix-multiplications in the summation, time complexity of  $B \times A$  is:

$$T(n)_{B \times A} = k \cdot \Theta(n^{\log_2 7}) = \Theta(k \cdot n^{\log_2 7})$$

## 2. Exercise 4.2-7 (Textbook page 83):

Assuming  $A = a + bi$  and  $B = c + di$ , then  $A.B$  is:

$$A.B = (ac - bd) + (ad + bc).i \implies \left\{ \begin{array}{l} \text{Real} = (ac - bd) \\ \text{Imag} = (ad + bc) \end{array} \right\}$$

Assuming  $u, v$ , and  $w$  as follows:

$$u = (a + b).(c + d) = ac + ad + bc + bd; // \text{ counts as 2 sum and 1 multiplication}$$

$$v = ac; // \text{ counts as 1 multiplication between } a \text{ and } c$$

$$w = bd; // \text{ counts as 1 multiplication between } b \text{ and } d$$

Then "Real" and "Imaginary" parts are:

$$\text{Real} = u - w; // \text{ counts as 1 sum}$$

$$\text{Imag} = u - v - w; // \text{ counts as 2 sum}$$

Thus, "Real" is computed with 1 summation and "Imag" is computed with 2 summations, and both share the same 3 multiplications. In total "Real" and "Imag" are computed with 3 multiplications and 3 summations.

## 3. Exercise 4.5-2 (Textbook page 97):

Considering Strassen's Algorithm, complexity of  $T(n) = 7T(\frac{n}{2}) + \Theta(n^2)$  is  $\Theta(n^{\log_2 7})$ . Since Strassen's algorithm is proven using CASE I of Master's theorem, it means that the Strassen recurrence is leave dominant.

The recurrence relation of Professor Caesar is  $T(n) = aT(\frac{n}{4}) + \Theta(n^2)$  and ths its complexity - basd on Strassen's algorithm - is:

$$T(n)_{Caesar} = \Theta(n^{\log_4 a})$$

If Professor Caesar's algorithm is better than Strassen algorithm, then its complexity must be smaller than the Sterassen's. In otherwords:

$$\begin{aligned} T(n)_{Caesar} < T(n)_{Strassen} &\implies \Theta(n^{\log_4 a}) < \Theta(n^{\log_2 7}) \implies n^{\log_4 a} < n^{\log_2 7} \implies \\ \log_4 a < \log_2 7 &\implies \log_2 \sqrt{a} < \log_2 7 \implies \sqrt{a} < 7 \implies a < 49 \checkmark \end{aligned}$$